Hybrid Dynamical Systems - Part I: Introduction

Sophie Tarbouriech

LAAS-CNRS, Toulouse, France
Common works with several colleagues

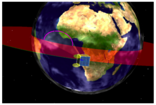
National
LAAS-CNRS, GIPSA-lab, LAGEP, CRAN, CRISTAL, LSS, ONERA, CHU-Toulouse

International
Brazil, Chile, Italy, Japan, Spain, Sweden, The Netherlands, UK, USA

January 20, 2023 - ISAE - Toulouse

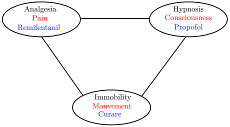
Outline

- Autonomous systems (aircraft, satellite): Aggressive maneuvering, High-precision control, Robust decision making

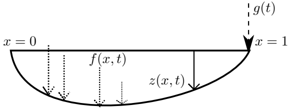


- Mechanical systems (robotic): High-precision control, high-speed control, vibration

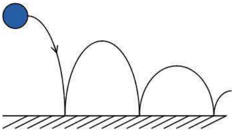
- Medical systems (anesthesia) : Drug injection for suspension of consciousness, pain and movement

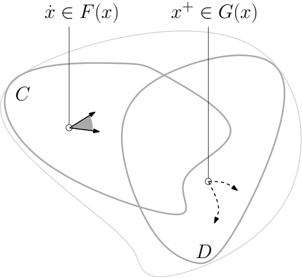


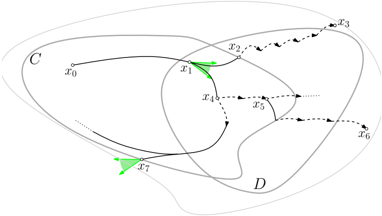
- Different kinds of models
  - ▷ Linear/nonlinear equations : ODE, PDE, algebro-differential



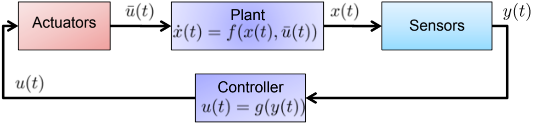
  - ▷ Different speed dynamics (slow/fast): singularly perturbed systems
  - ▷ Different nature of dynamics: hybrid dynamical systems, whose behavior is the results of a mix of differential/difference/logic equations







- ▷ Modification of the natural behavior of the system (which refers to the open-loop system): closed loop to satisfy some properties



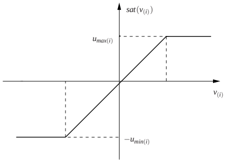
- ▷ Very simple example. Vertical position of the baby.

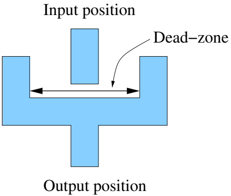
Indeed, baby tries to control its position (from the ground to stand up)

- baby = inverted pendulum
- diaper = a sort of uncertainty

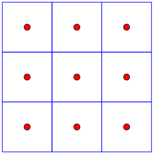
- The main objective resides in the capacity to ensure some properties (stability, performance, robustness, safety, tolerance, algorithmic convergence, ...), which are difficult or impossible to check analytically.
- The solution consists in developing indirect ways in order to ensure such properties of at least an avatar of this one sufficiently representative.

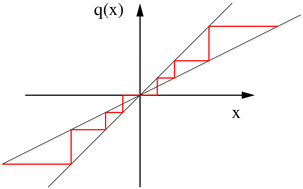
- Presence of various complex isolated elements in the control loop
  - ▷ nonlinearities as saturation, backlash, hysteresis, deadzone, discontinuous elements, ...



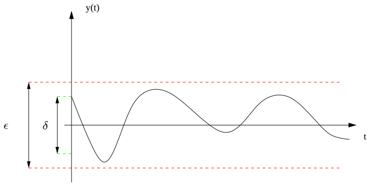


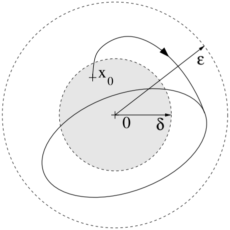
  - ▷ limited information in the communication channel: quantized input/output, encoder/decoder, sampled-data, ...



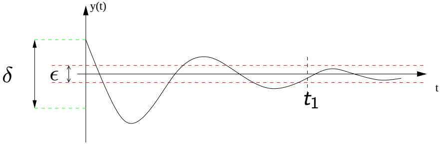


- A desirable **property** can be represented by the well known notions of stability and attractivity.
  - ▷ Stability characterizes the property that solutions starting close to some region (or set) remain close to that set.





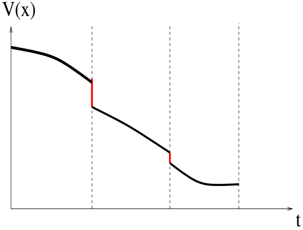
  - ▷ Attractivity characterizes the property that solutions starting away from this set will approach the set as time passes.

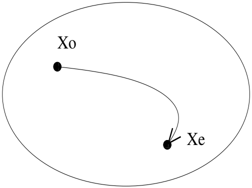


- Question: How can I check the stability (attractivity)?
  - ▷ Checking the two properties (stability and attractivity) is hard because it requires enumerating all possible evolutions, or solutions of the considered system
  - ▷ That appears to be intractable!
- Solution: use Lyapunov theory

Lyapunov theory (Aleksandr Mikhailovich Lyapunov (1857-1918))
Main tool to study the stability

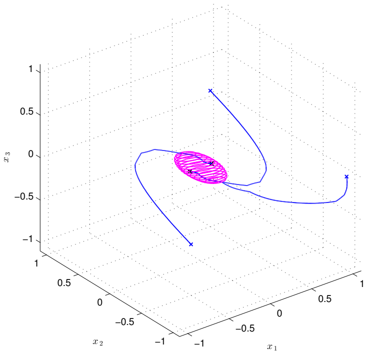
- ▷ The objective is to study the convergence of the system trajectories towards the origin (equilibrium point of interest) without explicit description of these trajectories.
- ▷ Stability certificate: Find a function with a particular sign $(V(x) > 0)$ and for which the time-derivative function (representing the decreasing) is also with a particular sign $(\dot{V}(x)/\Delta V(x) < 0)$

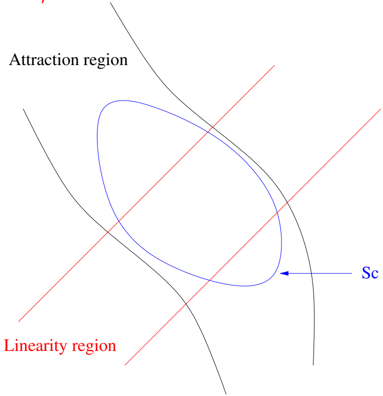




The notion of stability plays a major role for the analysis of dynamical systems or the design of control laws

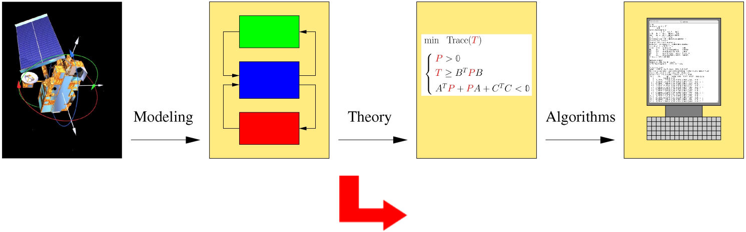
- Different kinds stability can be studied/obtained: global stability and regional (local) stability
- Important concepts: the notions of equilibrium points (ex: the origin), region de stability, region of attraction, attractor





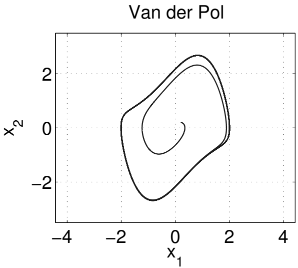
- Large class of systems of interest:
  - ▷ embedded systems
  - ▷ cyber-physical systems
  - ▷ heterogeneous components and interfaces (human, networks, digital devices)
- Applications:
  - ▷ aeronautical/aerospace
  - ▷ biological systems/Health
- Objectives: Analysis/Design of controllers under complex dynamics.
  - ▷ Stability/safety (characterization of the best region of operation)
  - ▷ Performance level in the presence of perturbation (e.g., wind gust)
  - ▷ Robustness in the presence of uncertainty on the system (e.g., variation of parameters)
  - ▷ Experimental validation/redesign when measurement not available.
- Main framework:
  - ▷ dedicated representations for isolated nonlinear elements
  - ▷ dynamical hybrid systems

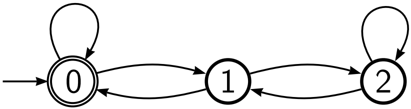
- Motivation: Develop tools in order to have a mathematical proof of the results
- Objective. Provide a mathematical certification of the validity of the properties and associate numerical procedure (constructive solutions).
- The method used can be summarized below



Constructive solutions

- In this lecture, we focus on hybrid dynamical systems: systems governed by dynamics of different natures: hybrid systems, whose the behavior is the results of a mix of differential/difference/logic equations





$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_1 &= -x_1 + x_2(1 - x_1^2)\end{aligned}$$

$$x^+ \in \begin{cases} \{0,1\} & \text{if } x = 0\\ \{0,2\} & \text{if } x = 1\\ \{1,2\} & \text{if } x = 2\end{cases}$$

- Example of such a kind of systems: circuits that combine analog and digital components; mechanical devices controlled by digital computers...

- It is interesting to recall the following:
  - ▷ Continuous-time dynamical systems are clearly described by differential equations:

$$\dot{x} = f(x) \text{ namely } \frac{dx(t)}{dt} = f(x(t)), \forall t \in \Re_{\geq 0}$$

    The solutions are continuous so that the integral of their derivative is the same function (Lebesgue measurability).
  - ▷ Discrete-time dynamical systems are commonly described by difference equations:

$$x^{+} = g(x) \text{ namely } x(j+1) = g(x(j)), \forall j \in N$$

    The solutions jump and therefore are discontinuous.
- The notation $\dot{x}$ represents the time derivative of x, while x^{+} is the value of the state after a discrete change.

- However, standard differential equations cannot describe changes of a logical variable that can take on only the values of 0 and 1.
  - ▷ That means that differential equations are not able to model a continuous-time system controlled by an algorithm involving logic.
  - ▷ Such a closed-loop system may be modeled through a combination of differential equations with difference equations.

- To summary, some examples of Hybrid Dynamical Systems:
  - ▷ Continuous systems with a phased operation:
    The systems for which the behavior changes along different phases, systems with switches, cyclic systems, systems with impact, ...
    Ex: Bouncing ball, walking robots, biological cell growth and division, billiard;
  - ▷ Continuous systems controlled by discrete logic:
    The revolution in digital technology has pointed out a need for design techniques that can guarantee safety and performance specifications of embedded systems, or systems that couple discrete logic with the analog physical environment.
    Ex: thermostat, chemical plants with valves, pumps, control modes for complex systems, eg. intelligent cruise control in automobiles, aircraft autopilot modes...

- To summary, some examples of Hybrid Dynamical Systems (cont'ed):
  - ▷ Coordinating processes:
    Systems which are comprised of many interacting subsystems (multi-agent systems) typically feature continuous controllers to optimize performance of individual agents, and coordination among agents to compete for scarce resources, resolve conflicts, etc.
    Ex: air and ground transportation systems, swarms of micro-air vehicles, evolution of opinions in social networks

- Recall that for both continuous and discrete-time systems the following concepts are well established:
    - ▷ Existence of solutions
    - ▷ Properties of stability (asymptotic, exponential, ...)
    - ▷ Uniform and global versions of them
    - ▷ Domains of attraction
    - ▷ Lyapunov functions and associated theorems
    - ▷ LaSalle invariance principles
    - ▷ Linearization
- Much effort has been taken in the past 20 years to extend (at least partially) these concept to the case of systems whose solutions are continuous but exhibit jumps: hybrid dynamical systems

Hybrid techniques reach beyond limits of classical control

- We will study a hybrid loop to augment the continuous-time controller by jump rules
  - ▷ For this we use reset control systems formalism [Prieur et al.,. 2018, Prieur et al. 2012] or, more generally hybrid dynamical systems formalism [Teel, Goebel, Sanfelice 2011]
- The use of hybrid systems is desirable due to their capability to:
  - ▷ provide global asymptotic stability of closed loops not stabilizable by continuous feedback (see e.g. [Hespanha et al, 1999, 2003]).
  - ▷ guarantee a robustness with respect to small errors in the loop, which cannot be obtained using classical (i.e. with a continuous dynamics) controllers (see e.g. [Prieur 2005, Goebel and Teel, 2009]).
  - ▷ improve the performance of linear systems in presence of disturbances [Becker et al, 2004, Nesic et al, 2008, Witvoet et al, 2007]
  - ▷ More generally, hybrid systems can model a wider range of physical problems and provide improved observers [Allgower et al. 2007, Prieur et al. 2012]

- First question: what is a hybrid dynamical system?
- Dynamical hybrid systems are then systems whose dynamics are governed by the combination of
  - a flow map of the type
$$\dot{x} = f(x, v) \tag{1}$$
    only active in certain subsets of the state space, called the flow set
  - and a jump map of the type
$$x^{+} = g(x, v) \tag{2}$$
    which is active in another subset of the state space, called the jump set.

- The hybrid system can be described as follows:

$$\begin{aligned} \dot{x} &= f(x) \text{ if } x \in \mathcal{C} \\ x^+ &= g(x) \text{ if } x \in \mathcal{D} \end{aligned} \tag{3}$$

  where

  - f describes the continuous evolution (flow) of the state
  - g describes the discrete evolution (jumps) of the state
  - $\mathcal{C}$ describes the set where the flow can occur (continuous evolution)
  - $\mathcal{D}$ describes the set where the jumps can occur (discrete evolution)

- The sets $\mathcal{C}$ and $\mathcal{D}$ may represent

  - physical constraints,
  - model for logical modes by constraining them to some discrete set (for example $0, 1$ for "off" and "on"),
  - switching law, ...

- From the hybrid formalism and from the following equations

$$\begin{array}{c} \dot{x} = f(x) \text{ if } x \in \mathcal{C} \\ x^{+} = g(x) \text{ if } x \in \mathcal{D} \end{array} \tag{4}$$

  we can describe pure:

  ▷ continuous-time dynamical systems if $\mathcal{C} = \Re^n$ and $\mathcal{D} = \emptyset$

  ▷ or discrete-time dynamical systems if $\mathcal{C} = \emptyset$ and $\mathcal{D} = \Re^n$.

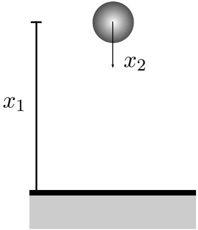
- System (3) can be considered as a set of constraints satisfied by a solution but for which multiple solutions may occur, in particular if $\mathcal{C} \cap \mathcal{D} \neq \emptyset$.
- Then, in this case, it is useful to look at a more general description with

$$\begin{array}{c} \dot{x} \in F(x) \text{ if } x \in \mathcal{C} \\ x^+ \in G(x) \text{ if } x \in \mathcal{D} \end{array} \tag{5}$$

  ▷ Both F and G may be set-valued to represent effects of perturbations, uncertainty, lack of determinism, decision-making capabilities, overlapping guards and different resets, ...

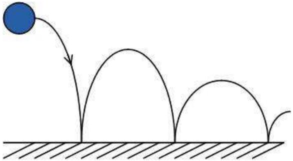
A first example: bouncing ball (1)

- Consider a ball with unit mass, height $p = x_1 \in \mathbb{R}$ and vertical velocity $v = x_2 \in \mathbb{R}$, for which the only external force acting on the ball is gravity.
- The ball is dropped from some height above the floor
- A simplified way to model the behavior of the ball is
  - ▷ to assume that only gravity affects the ball when the ball is above the floor,
  - ▷ and that the collision of the ball with the floor produces an instantaneous effect on the velocity of the ball.
- In such a model, the velocity of the ball evolves continuously above the floor until a collision occurs.



A first example: bouncing ball (2)

- During a collision, the velocity undergoes a discontinuous change. More precisely, when the ball hits the ground, its velocity is instantaneously reversed and the ball takes off again.
- After each collision, and so after each jump in the velocity, the ball position and velocity evolve continuously again, until the next collision with the floor.
- The state of the point-mass can be described with $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathbb{R}^2$



x_1 represents the height above the surface and x_2 represents the vertical velocity.

A first example: bouncing ball (3)

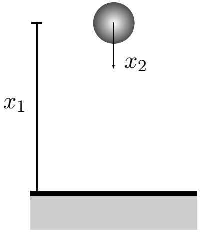
- The motion of the ball when $x_1 > 0$ is given by Newton's law:

$$\ddot{x}_1 = -\gamma, \ \gamma > 0$$

where γ is the acceleration due to gravity.

- Define $x_2 = \dot{x}_1$, one gets in the state-space form:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\gamma \end{aligned}$$



- During flows, the system evolves as a simple double integrator and flow is allowed only when the ball is above the ground, i.e. for positive values of x_1 .
- When the ball hits the ground, its velocity is instantaneously reversed and the ball takes off again. This suggests that jumps should be triggered only when $x_1 = 0$ and $x_2 < 0$, i.e. when the ball hits the ground and the velocity pierces the ground.
- The height does not change at collisions: $x_1^+ = x_1 = 0$.

A first example: bouncing ball (4)

- An hybrid model for the bouncing ball is given by

$$\dot{x} = f(x) = \begin{bmatrix} x_2 \\ -\gamma \end{bmatrix} \text{ if } x \in \mathcal{C} = \{x \in \mathbb{R}^2; x_1 > 0\}$$
$$x^+ = g(x) = \begin{bmatrix} 0 \\ -x_2 \end{bmatrix} \text{ if } x \in \mathcal{D} = \{x \in \mathbb{R}^2; x_1 = 0 \text{ and } x_2 < 0\}$$

  - ▷ With such definitions of the flow and jump sets, the ball is never allowed to reach the ground.
  - ▷ In fact, the set of points $\{x \in \mathbb{R}^2; x_1 = 0, x_2 \geq 0\}$ does not belong neither to $\mathcal{C}$ nor to $\mathcal{D}$

A first example: bouncing ball (5)

- The issue above mentioned can be solved by defining $\mathcal{C}$ and $\mathcal{D}$ as closed sets, i.e. using non-strict inequalities.
  - ▷ Recall that a set $\mathcal{S}$ is closed if its complement is an open set, and a set $\mathcal{S}$ is open if any point in $\mathcal{S}$ has a neighborhood fully contained in $\mathcal{S}$.
- In the rest of the lecture we will mainly consider the flow and jump sets to be closed.
- It is important mentioning that closed sets are one of the key ingredients for stability and robustness analysis of hybrid dynamical systems.

A first example: bouncing ball (6)

- We can now extend the model in order to capture the impact dynamics by means of a restitution factor $\lambda \in [0, 1]$
  - ▷ If $\lambda = 1$, the impact is fully elastic,
  - ▷ while $\lambda = 0$ denotes a completely non-elastic bounce.
- Then, the data of the hybrid system is the following:

$$\dot{x} = f(x) = \begin{bmatrix} x_2 \\ -\gamma \end{bmatrix} \text{ if } x \in \mathcal{C} = \{x \in \Re^2; x_1 \geq 0\}$$

$$x^+ = g(x) = \begin{bmatrix} 0 \\ -\lambda x_2 \end{bmatrix} \text{ if } x \in \mathcal{D} = \{x \in \Re^2; x_1 = 0 \text{ and } x_2 \leq 0\}$$

A first example: bouncing ball (7)

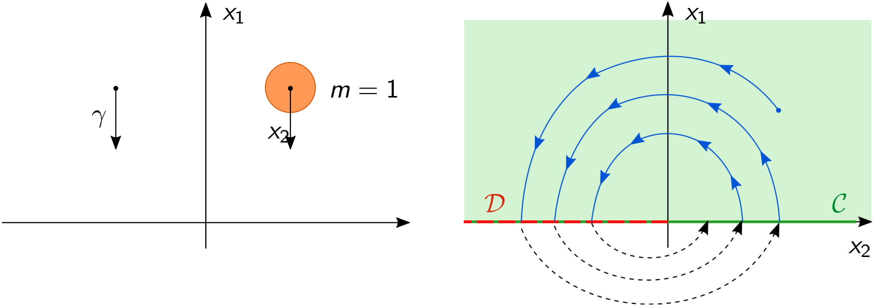


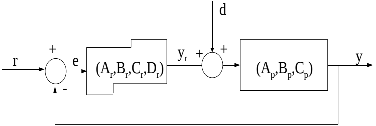
Figure: Bouncing ball example: problem definition and example of a solution.

A first example: bouncing ball (8)

- In principle, a hybrid dynamical system can present multiple solutions, depending
  - on the nature of the dynamics
  - and on the structure of the flow and jump sets.
- In general, solutions from $\mathcal{C} \cap \mathcal{D}$ (if this set is nonempty) can choose to either flow or jump, provided that they don't escape $\mathcal{C} \cup \mathcal{D}$.
- The concept of solution will be introduced in detail later in the course, but for the moment it is enough to notice that solutions to the bouncing ball system are unique.
  - In fact, from any point in $\mathcal{C} \cap \mathcal{D}$, solutions are only allowed to jump otherwise they would escape the flow set.
  - As a special case, solutions from the origin can only jump forever and remain in the origin; such a point belongs also to the flow set, but no solution can flow from there, while remaining in $\mathcal{C}$, even for an arbitrary small time.

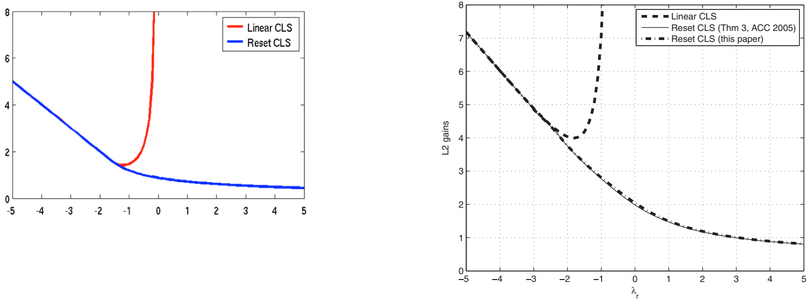
- Another interesting sub-class of dynamical hybrid systems:
  - ▷ The reset control systems.
  - ▷ A reset controller is a linear controller whose output is reset to zero whenever its input and output satisfy an appropriate algebraic relationship.
- Reset controllers reach beyond the use of classical linear and nonlinear control schemes because the state response of the closed-loop is a discontinuous function of time (due to the occurrence of resets).
  - ▷ Possible difficulty for the analysis of stability and performance,
  - ▷ However, that may allow in certain cases to achieve performance specifications that overcome the intrinsic limitations of classical control architectures.

Example: Reset controller (first order) controlling a linear system



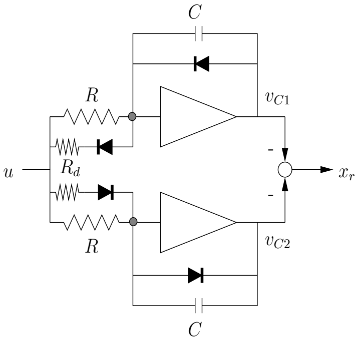
$$\begin{aligned} \dot{x}_r &= \lambda_r x_r + e \text{ if } x_r e \geq 0 \\ x_r^+ &= 0 \text{ if } x_r e \leq 0 \end{aligned}$$

$P(s) = \frac{1}{s}$ $\qquad$ $P(s) = \frac{s+1}{s(s+2)}$



L2 gain (between d and y) in function of λ_r [Zaccarian, Nesìc, Teel, 2011]

- Two examples: Clegg integrator and FORE
- Clegg integrator: [J.C. Clegg, 1958], [Chen, Hollot, Chait, 2000]



- ▷ It is a modification of an analog circuit which has the same behavior as a linear integrator when the input and the output of the circuit have the same signs ($x_r u \geq 0$).
- ▷ Otherwise, the output integrator is reset to zero ($x_r u \leq 0$).

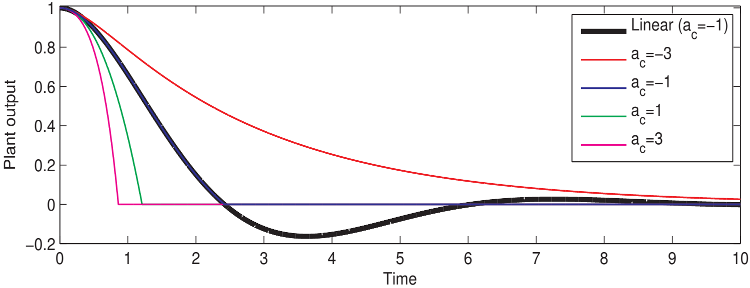
$$
\begin{aligned}
\dot{x}_r &= \frac{1}{RC} u \text{ if } x_r u \geq 0 \\
x_r^+ &= 0 \text{ if } x_r u \leq 0
\end{aligned}
$$

- First Order Reset Element (FORE): [Horowitz and Rosenbaum, 1975], [Chen, Hollot, Chait, 2000]
  - ▷ It corresponds to first order linear systems whose state is reset to zero whenever the input and the state values have opposite signs.

$$\begin{aligned} \dot{x}_r &= \lambda_r x_r + e \text{ if } x_r e \geq 0 \\ x_r^+ &= 0 \text{ if } x_r e \leq 0 \end{aligned}$$

  - ▷ FORE was first introduced as a generalization of the so-called Clegg integrator, which is the special case of a FORE having its pole at the origin $(\lambda_r = 0)$.

- One considers the plant controlled by the FORE: $P(s) = 1/s$ and different values for λ_r



- Interpretation: Resets remove overshoots, instability ($\lambda_r > 0$) improves transient

- An interesting class of hybrid systems resides in the class of systems with discrete states or logical modes.
- Often control systems involve a finite numbers of operating conditions, which can be enumerated:
  - ▷ Example: different controllers which switch, different gears in a car, ...
- The fact that a variable (say q) belongs to a discrete set means that it cannot flow and it only changes via jumps. The rest of the state may flow and also jump
  - ▷ Example: In thermostats, for example, the logical state will be 0 for "off" and 1 for "on" on while temperature evolves without jumps See next slide
- Another classes: hybrid automata (see the example 1.10, p.18 in the book [Goebel et al., 2012]), impulsive systems, switching systems, ...

A second example: Thermostat -Temperature regulation

- The idea through this example is to consider dynamical systems involving discrete logic states
  - ▷ That can be modeled as hybrid dynamical systems.
- In this example, we consider the on/off control of a thermostat to regulate the temperature inside a room.
  - ▷ The temperature evolves continuously and its dynamics can be modeled as a differential equation,
  - ▷ while the thermostat is turned on and off instantaneously, causing the logic variable to change only at jumps.
- Assume that the temperature in a room is z.

A second example: Thermostat -Temperature regulation

- The temperature evolves continuously and its dynamics can be modeled as a differential equation

$$\dot{z} = -(z - z_0) + z_d q = f(z, q)$$

  - ▷ z_0 is the equilibrium temperature in the room (natural one).
  - ▷ z_d is the efficiency of the heater
  - ▷ q is the state of the heater, which is either 1 ("on") or 0 ("off")
- Typically, one wants to stabilize z between z_{min} and z_{max} such that:

$$z_0 < z_{min} < z_{max} < z_0 + z_d$$

A second example: Thermostat -Temperature regulation

Remark

One gets:

- for $q = 1$ one clearly satisfies $f(z, 1) = -z + z_0 + z_d > 0$ for $z \leq z_{max}$ (the heater indeed heats, while it is on)
- for $q = 0$ one gets $f(z, 0) = -z + z_0 < 0$ for $z \geq z_{min}$ (the temperature cools down when the heater is off).

A second example: Thermostat -Temperature regulation

- Indeed, the thermostat position q is turned on and off instantaneously, causing the logic variable to change only at jumps.
- The idea is to introduce a hysteresis mechanism
  - ▷ We use a logic variable $q \in \{0, 1\}$ to implement a hysteresis mechanism
  - ▷ it turns off the heater $(q^+ = 0)$ when the temperature z exceeds a given threshold z_{max}
  - ▷ it turns it on the heater $(q^+ = 1)$ when z is smaller than a given z_{min}.

A second example: Thermostat -Temperature regulation

- Recall that the state of the complete systems is then $x = (z, q) \in \mathbb{R} \times \{0, 1\}$.
- Then with $Q = \{0, 1\}$, one obtains the following hybrid systems:

$$\begin{aligned} \dot{z} &= -(z - z_0) + z_d q \\ \dot{q} &= 0 \end{aligned} \quad , (z, q) \in \mathcal{C} = \bigcup_{q \in Q} (\{q\} \times C_q)$$

$$\begin{aligned} z^+ &= z \\ q^+ &= 1 - q \end{aligned} \quad , (z, q) \in \mathcal{D} = \bigcup_{q \in Q} (\{q\} \times D_q)$$

- The sets C_q and D_q are defined by:

The jump set controls the logic of when the heater should turn on and off

$$C_0 = \{z; z \geq z_{min}\}, C_1 = \{z; z \leq z_{max}\}, D_0 = \{z; z \leq z_{min}\}, D_1 = \{z; z \geq z_{max}\}$$

A second example: Thermostat -Temperature regulation

- Initial condition: $z < z_{min}$ and $q = 1$,
- The solution flows until the upper limit z_{max} is reached.
- Once z reaches z_{max}, the solution cannot flow anymore, but since this point is in the jump set, then $q = 0$ and the temperature starts dropping.

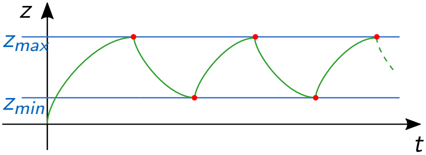


Figure: Temperature evolution as a function of continuous time t.

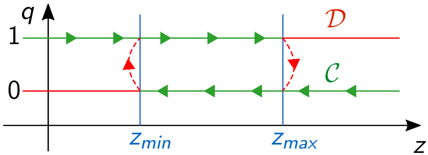


Figure: State space representation.

A second example: Thermostat -Temperature regulation

Remarks

- z is kept within the desired range $[z_{min}, z_{max}]$
  - ▷ This is a bounded and closed set, namely a compact set.
- This mechanism generates a hybrid limit cycle
  - ▷ solutions converge to (asymptotically or in finite time, as in this case) a periodic trajectory.

- Suggestion. Study the way to extend the modeling framework to the broader class of systems having logical modes, where a logic state variable $q \in \{1, 2, ..., N-1\}$ takes values in finite set of N possible modes (in the thermostat there are only two modes $N = 2$).

Example 3: Sample-and-Hold Feedback Control

- In this example, the state z of a continuous-time analog plant is measured every T seconds and used as input to a digital controller.
- Then, at every sampling time, the control law $\kappa(z)$ is instantaneously updated and applied at the input of the plant by means of a zero-order hold (ZOH).

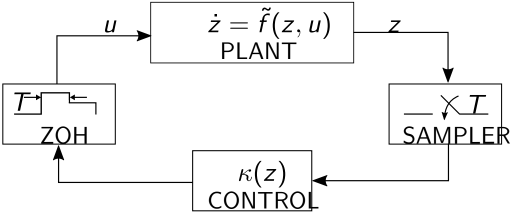


Figure: Block diagram of the sample-and-hold feedback control.

Example 3: Sample-and-Hold Feedback Control

- The sampler and the ZOH can be regarded as interfaces between the continuous-time and discrete-time parts of the system.
  - ▷ We will model the sampler as a timer τ which is reset every T seconds,
  - ▷ The ZOH is modeled as a memory element updated at every sampling time

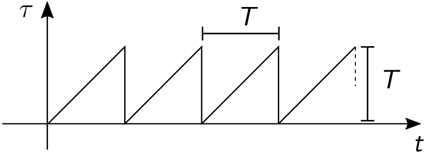


Figure: Timer state τ as function of time.

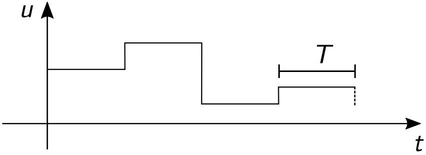


Figure: Memory state u as function of time.

Example 3: Sample-and-Hold Feedback Control

- Given a positive scalar T, the state of the sampled-data system can be defined as $x = (z, u, \tau) \in \mathbb{R}^{n_p} \times \mathbb{R}^{n_u} \times [0, T]$, where n_p, n_u are the dimensions of the plant and controller states, respectively.
- The resulting hybrid dynamical system is modeled as follows:

$$\dot{x} = \begin{bmatrix} \dot{z} \\ \dot{u} \\ \dot{\tau} \end{bmatrix} = f(x) := \begin{bmatrix} \tilde{f}(z,u) \\ 0 \\ 1 \end{bmatrix}, \quad x \in \mathcal{C} := \{(z,u,\tau) : \tau \in [0,T]\}$$

$$= \mathbb{R}^{n_p} \times \mathbb{R}^{n_u} \times [0, T],$$

$$x^+ = \begin{bmatrix} z^+ \\ u^+ \\ \tau^+ \end{bmatrix} = g(x) := \begin{bmatrix} z \\ \kappa(z) \\ 0 \end{bmatrix}, \quad x \in \mathcal{D} := \{(z,u,\tau) : \tau = T\}$$

$$= \mathbb{R}^{n_p} \times \mathbb{R}^{n_u} \times \{T\}.$$

Example 3: Sample-and-Hold Feedback Control

- At every sampling time T, the timer triggers a jump:
  - ▷ τ is reset to 0,
  - ▷ while the control law u is updated according to the function $\kappa(z)$.
- Since we are considering the plant to be a physical system, its evolution is continuous and is not affected by jumps.
- During flows, the timer increases linearly and the control signal is constant (its time derivative is equal to zero).
- As in the example of the bouncing ball (Example 1), the flow and jump sets are closed.

Example 3: Sample-and-Hold Feedback Control

- Let us consider a more realistic scenario:
  - ▷ The timer resets is affected by jitter, i.e. uncertainty (perturbing the periodicity of the sampling)
  - ▷ In this case, the value of τ after a jump is not always 0 but can take any value in a small interval $[-\varepsilon, \varepsilon]$.
- This behavior leads to non-uniqueness of solutions and can be modeled as a difference inclusion:

$$x^+ \in G(x) := \begin{bmatrix} z \\ \kappa(z) \\ [-\varepsilon, \varepsilon] \end{bmatrix}, \quad x \in \mathcal{D}. \tag{6}$$

Suggestion - Example 4: quantized systems

- See Example 1.5 in the book of Teel.
- The plant is defined by:
$$\dot{x} = x + u \ .$$
- The controller $u = -2x$ succeeds to stabilize the system.
- Consider a quantizer satisfying $\forall z \in \mathbb{R}$, $|z| \leq M$,
$$|q(z) - z| \leq \Delta \ ,$$
with $M = 10$ and $\Delta = 1$, and satisfying $\forall z \in \mathbb{R}$, $|q(z)| \leq M - \Delta$,
$$|z| \leq M \ .$$
- The hybrid system results from the interconnection of the plant with the hybrid controller:
$$\begin{array}{c} u = -2q(\frac{x}{\mu}) \\ \dot{\mu} = 0 \ , \quad (x,\mu) \in C \\ \mu^+ = \lambda_{in}\mu \ , \quad (x,\mu) \in D_{in} \\ \mu^+ = \lambda_{out}\mu \ , \quad (x,\mu) \in D_{out} \end{array}$$
with $\lambda_{in} = 0.5$ and $\lambda_{out} = 1.5$.

- In this lecture, we considered the class of dynamical hybrid systems, which combine continuous and discrete-time dynamics.
- We have described the system under consideration
  - ▷ The description of the flow and jump sets is very important
- We have considered several examples in order to fix the ideas
- Several important notions will studied in the next lecture: time-domain, solutions, ... details are clearly given in the book of Teel.
- Main sources for this lecture: Chapter 1 of the book of Teel, notes of Luca Zaccarian, Christophe Prieur, Francesco Ferrante, Ricardo Sanfelice, ST

- S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. Linear Matrix Inequalities in System and Control Theory. SIAM Studies in Applied Mathematics, Philadelphia, PA, 1994.
- M. Fiacchini, S. Tarbouriech, C. Prieur. Quadratic stability for hybrid systems with nested saturations, IEEE Transactions on Automatic Control, vol.57, no.7, pp.1832-1838, 2012.
- F. Fichera, C. Prieur, S. Tarbouriech, L. Zaccarian. LMI-Based Reset H_∞ Design for Linear Continuous-Time Plants, IEEE Transactions on Automatic Control, Volume 61, Issue 12, pp.4157-4163, december 2016.
- R. Goebel, R. Sanfelice and A. R. Teel. Hybrid dynamical systems. IEEE Control Systems Magazine 29(2), 28-93, 2009.
- R. Goebel, R. Sanfelice and A. R. Teel. Hybrid dynamical systems. Modeling, stability and Robustness, Princeton University Press, 2012.
- F. Ferrante, F. Gouaisbaut, R. Sanfelice, S. Tarbouriech. State estimation of linear systems in the presence of sporadic measurements, Automatica, Volume 73, pp.101-109, November 2016,

- L. Hetel, J. Daafouz, S. Tarbouriech, C. Prieur. Stabilization of linear impulsive systems through a nearly-periodic reset, Nonlinear Analysis: Hybrid Systems, Vol.7, N¡1, pp.4-15, 2013.
- H. K. Khalil. Nonlinear Systems. Prentice Hall, Third Edition, 2002.
- T. Loquen, S. Tarbouriech and C. Prieur. Stability of reset control systems with nonzero reference. In: 47th IEEE Conference on Decision and Control (CDC). Cancun, Mexico. pp. 3386-3391, 2008.
- A. M. Lyapunov. The general problem of the stability of motion. International Journal of Control, 55(3):531-534, 1992.
- D. Nesic, L. Zaccarian and A.R. Teel. Stability properties of reset systems. Automatica 44(8), 2019-2026, 2008.
- C. Prieur, S. Tarbouriech, L. Zaccarian. Lyapunov-based hybrid loops for stability and performance of continuous-time control systems, Automatica, vol. 49, 2, pp. 577-584, 2013.
- R. Sanfelice. Lecture on Hybrid System, HYCON2, Paris, 2014.
- Software: https://hybrid.soe.ucsc.edu/software