Georgia Institute of Technology

Dr. Magnus Egerstedt
Professor
School of Electrical and
Computer Engineering

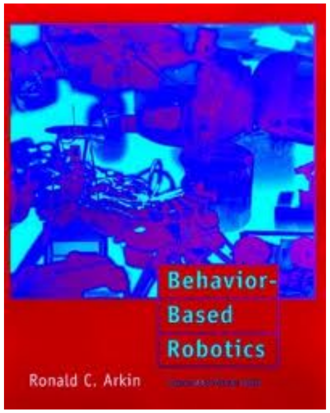
Control of Mobile Robots

Module 5
Hybrid Systems

How make mobile robots move in effective, safe, predictable, and collaborative ways using modern control theory?

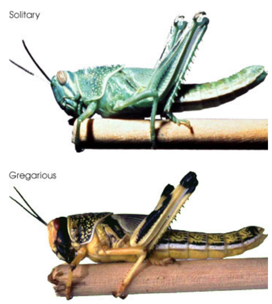
Lecture 5.1 – Switches Everywhere

- So far, the models stay the same over time
  - NOT ALWAYS TRUE!
- We have a designed one-size-fits-all controllers
  - "NEVER" TRUE IN ROBOTICS!



- Recall the behavior-based control paradigm
- We need to be able to deal with these types of phenomena

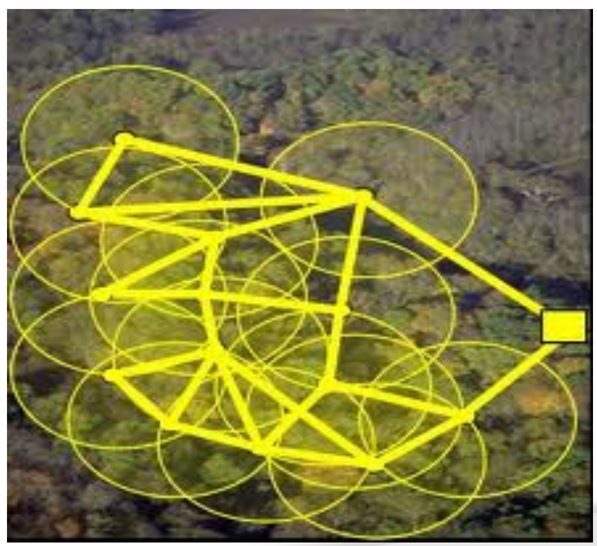
Switches by Necessity



bipedal robot

Switches by Design

for the better not because we have to



Switches by Design

Issues

- Models?
- Stability and Performance? $(t \not\to \infty)$
- Compositionality?
- Traps?

↳ t is never stable

↳ Put all controllers together

Lecture 5.2 – Hybrid Automata

- We need to be able to describe systems that contain both the continuous dynamics and the discrete switch logic
- ***Hybrid Automata*** = Finite state machines (discrete logic) on steroids (continuous dynamics)

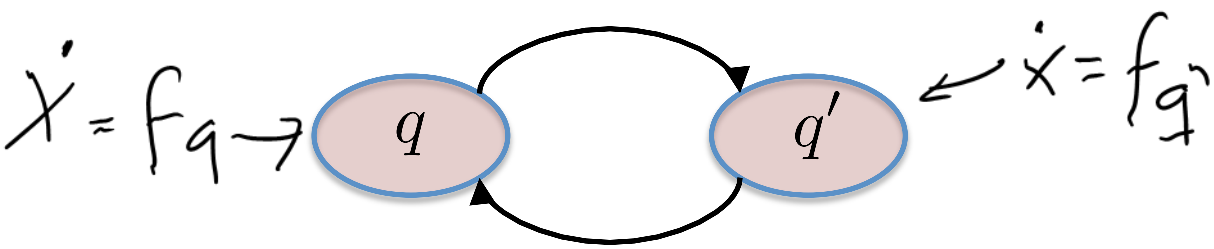
↳ multiple controllers

Modes, Transitions, Guards, and Resets

- Let, as before, the (continuous) state of the system be x
- As we will be switching between different *modes of operation*, let's add an additional discrete state q which mode I am in
- Dynamics:

$$\dot{x} = f_q(x, u)$$

- The *transitions* between different discrete modes can be encoded in a state machine:



how it affects the state

→ what states we are moving

Modes, Transitions, Guards, and Resets

↳ individual dynamics ↳ when to make the jump

- The conditions under which a transition occurs are called guard *conditions*, i.e., a transition occurs from q to q' if

$$x \in G_{q,q'}$$

Guard (when in time to jump)

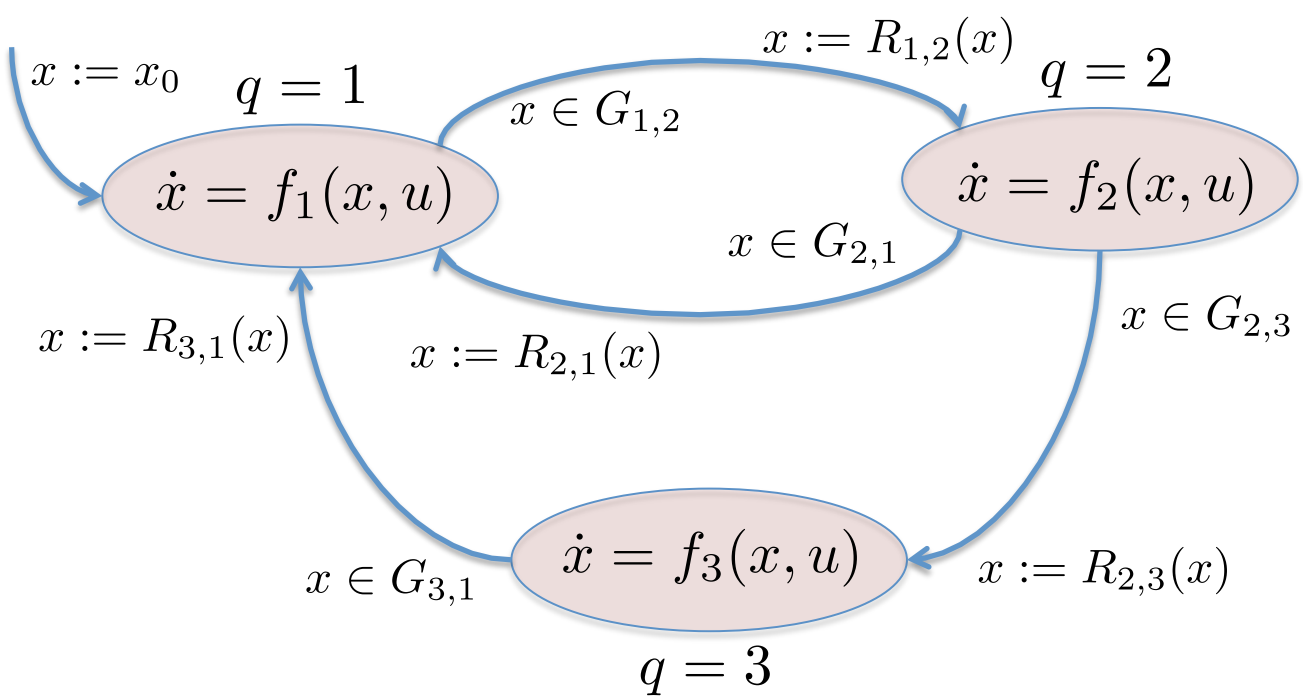
- As a final component, we would like to allow for abrupt changes in the continuous state as the transitions occur, which we will call *resets*:

Reset

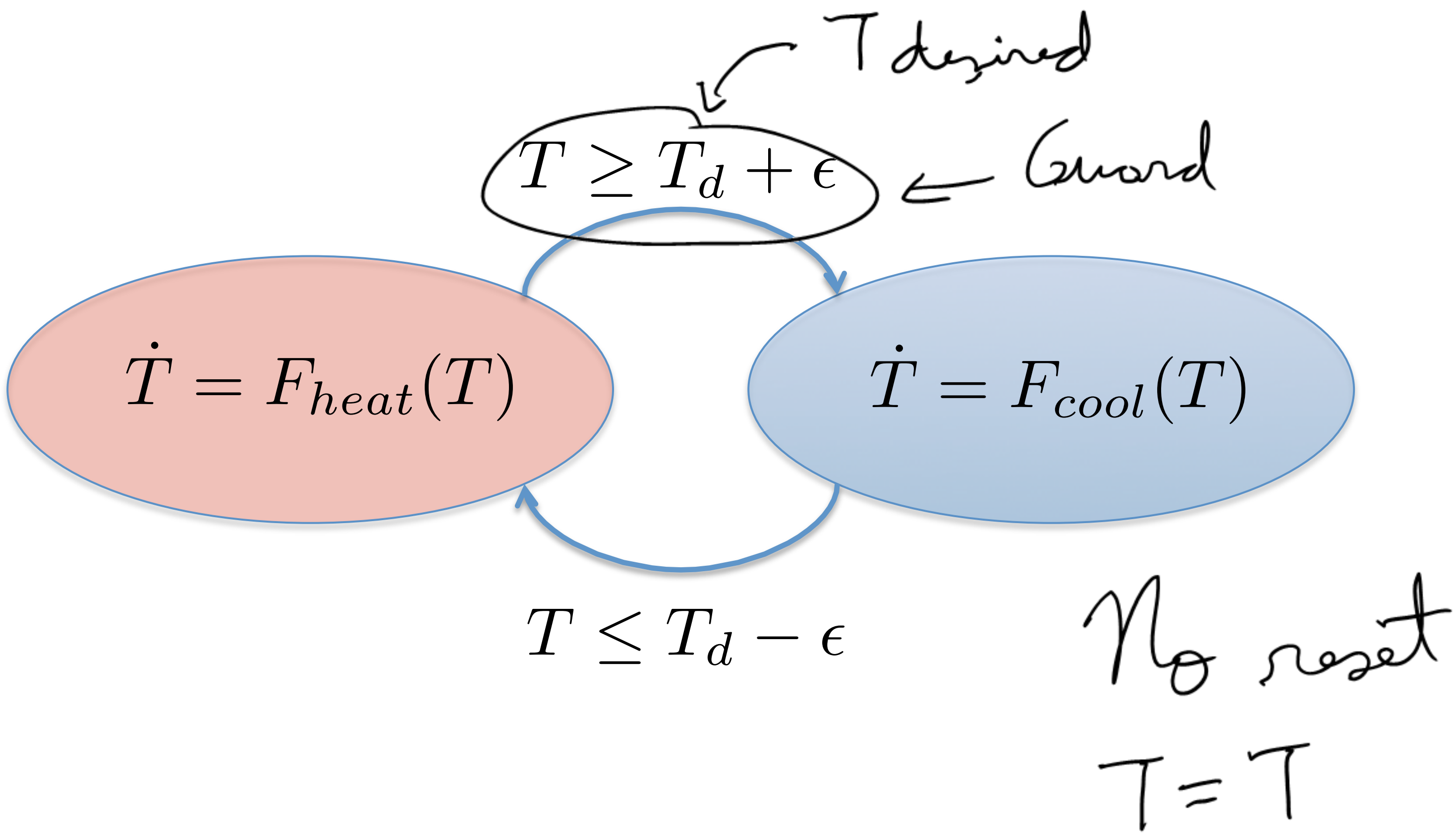
$$x := R_{q,q'}(x)$$

The Hybrid Automata Model

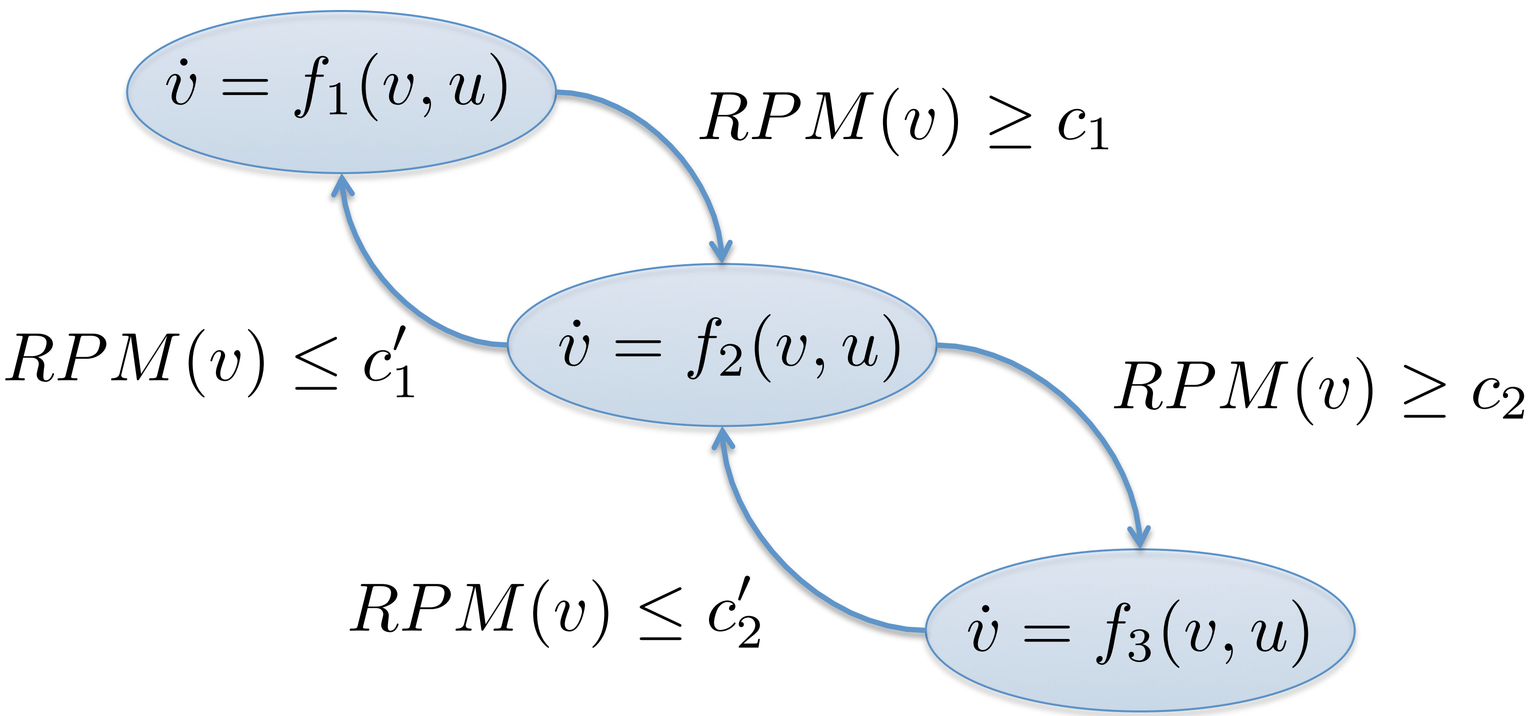
- Putting all of this together yields a very rich model known as a hybrid automata (HA) model:



HA Example 1 - Thermostat



HA Example 2 – Gear Shift



C_1 has to be different to C_1' so we 2 guards are not true at the same time

HA Example 3 – Behaviors

$$\|x - x_O\| \leq d$$

if obstacle is close

$$\dot{x} = f_{GTG}(x) \qquad \dot{x} = f_{AO}(x)$$

goal to goal

Avoid obstacle

$$\|x - x_O\| \geq d'$$

if obstacle is far

Lecture 5.3 – A Counter Example

- What can possibly go wrong when you start switching between different controllers?

A Simple 2-Mode System

- Two modes:

$$\dot{x} = A_1 x = \begin{bmatrix} -\epsilon & 1 \\ -2 & -\epsilon \end{bmatrix} x$$

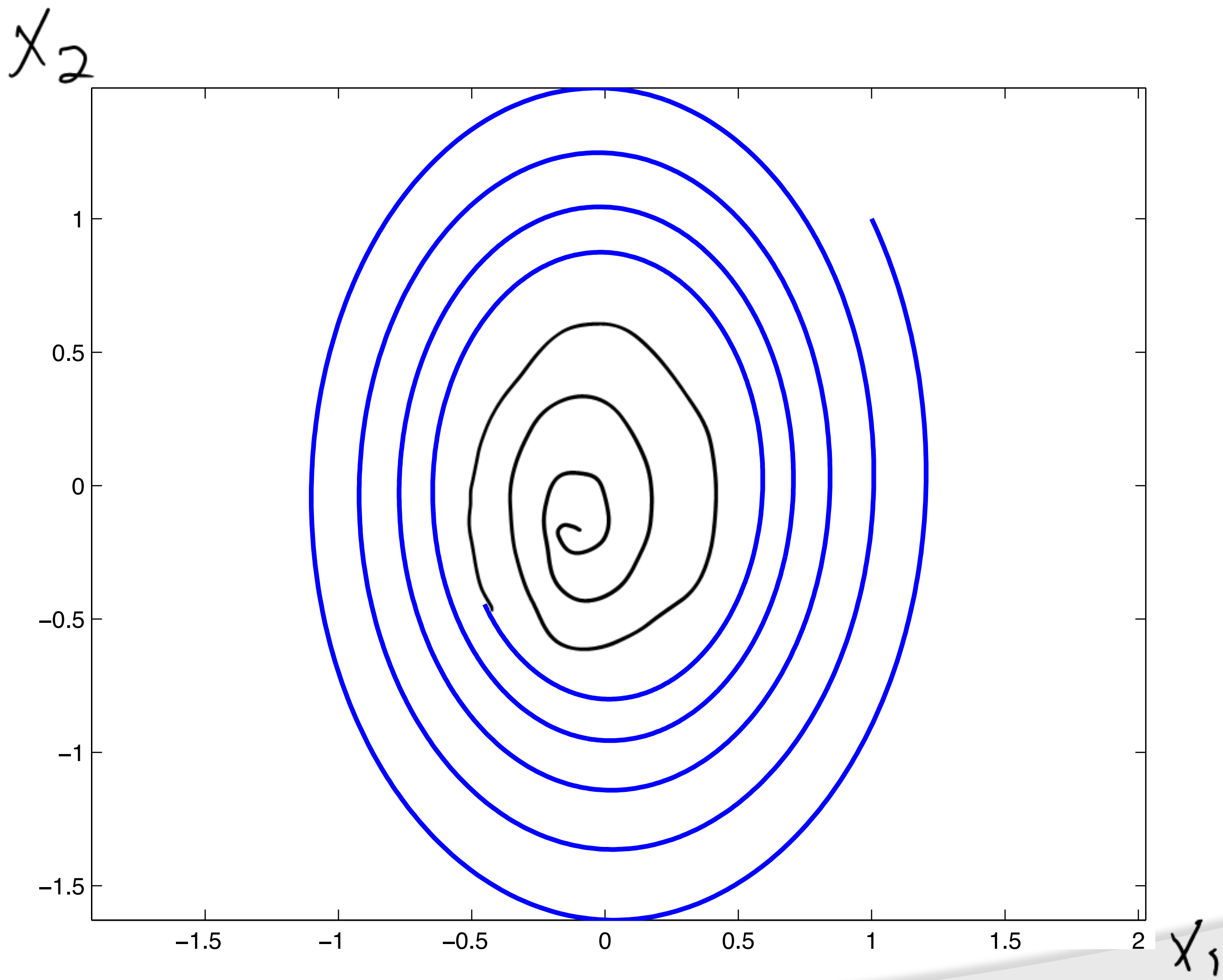
some
oscillation

$$\text{eig}(A_i) = -\epsilon \pm 1.41j$$

$$\dot{x} = A_2 x = \begin{bmatrix} -\epsilon & 2 \\ -1 & -\epsilon \end{bmatrix} x$$

- Both modes are asymptotically stable!

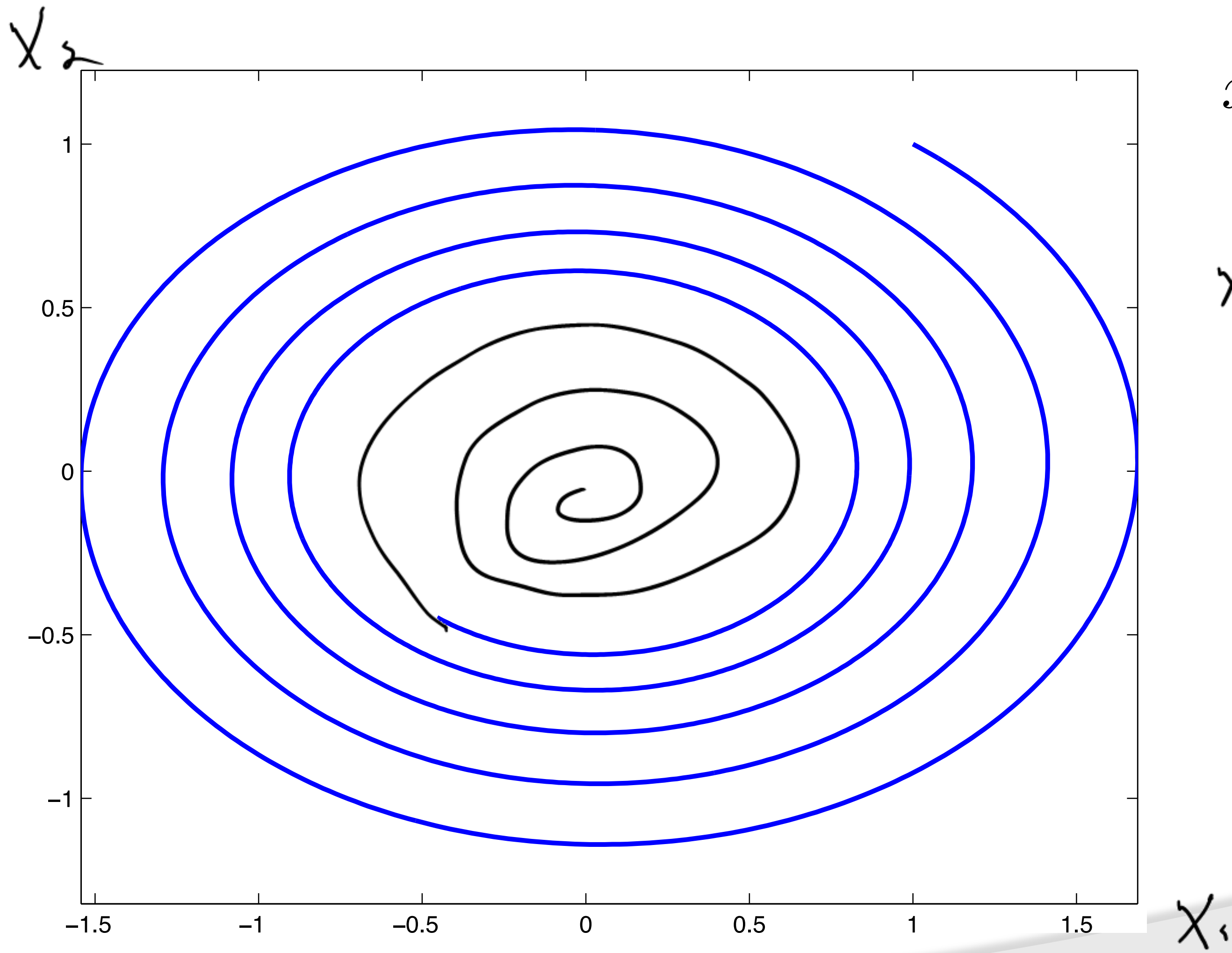
Mode 1



$$\dot{x} = A_1 x$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Mode 2



$$\dot{x} = A_2 x$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

HA 1 ← hybrid automata

$x_2 = 0$

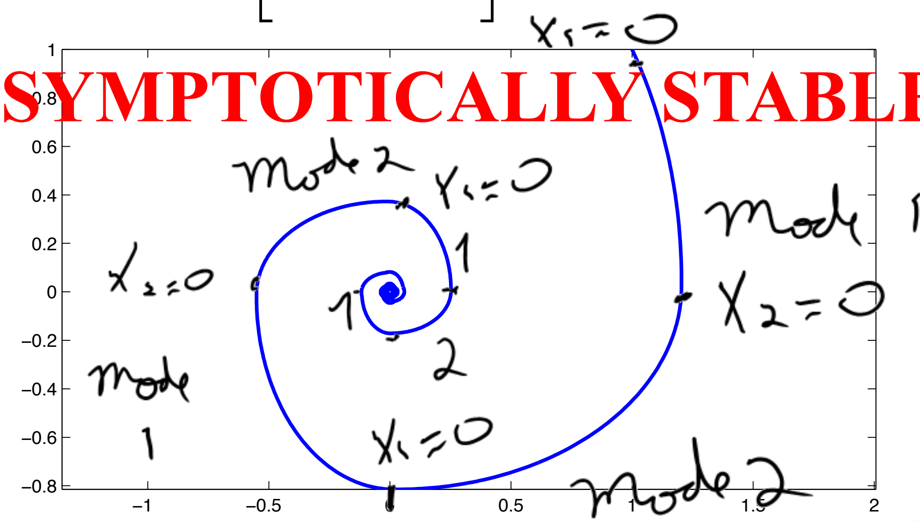
$$\begin{bmatrix} 0 & 1 \end{bmatrix} x = 0$$

$$\dot{x} = A_1 x \qquad \dot{x} = A_2 x$$

$$\begin{bmatrix} 1 & 0 \end{bmatrix} x = 0$$

$x_1 = 0$

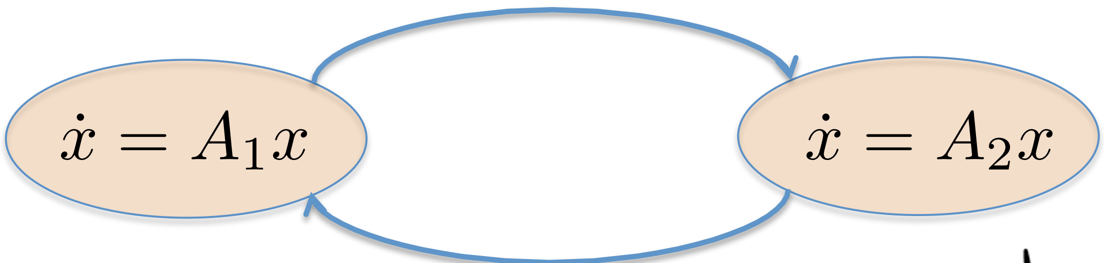
ASYMPTOTICALLY STABLE



HA 2

$x_1 = 0$

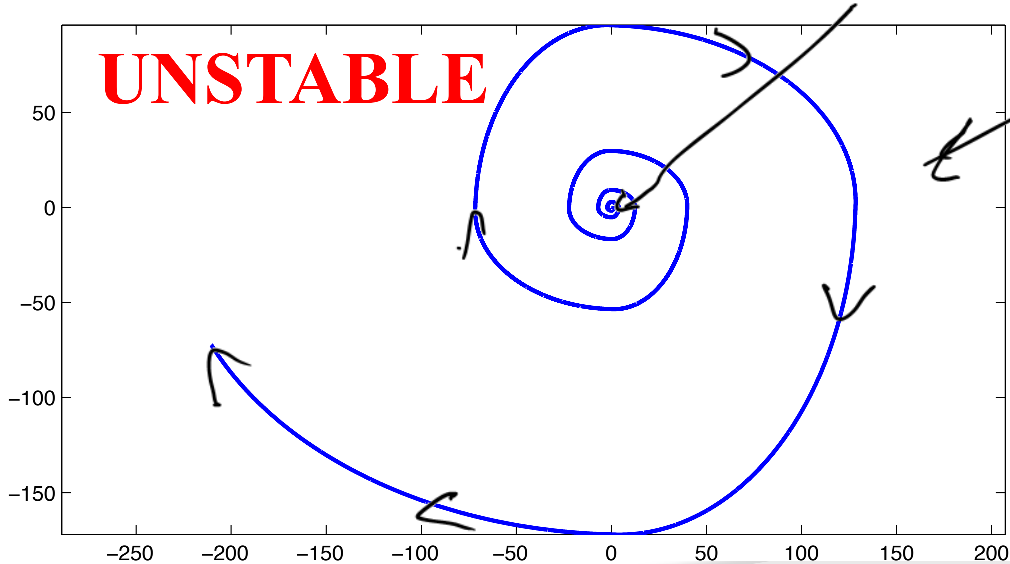
$$\begin{bmatrix} 1 & 0 \end{bmatrix} x = 0$$ Guard



$x_2 = 0$ $$\begin{bmatrix} 0 & 1 \end{bmatrix} x = 0$$

start point

spiral out



Punchlines

- By combining stable modes, the resulting hybrid system may be unstable!
- By combining unstable modes, the resulting hybrid system may be stable!
- Design stable modes but be aware that this is a risk one may face!

↳ it will be good for design to check this

Lecture 5.4 – Danger, Beware!

- Stable subsystems do not guarantee a stable hybrid system

Switched Systems

- We saw last time that it was possible to destabilize stable subsystems by an unfortunate series of switches
- Ignoring resets, we can write the hybrid system as a *switched system*:

$$\dot{x} = f_{\sigma}(x, u)$$

switch signal

- The *switch signal* dictates which discrete mode the system is in

$$\sigma(t) \in \Sigma = \underbrace{\{1, \ldots, p\}}_{\text{modes}}$$

Different Kinds of Stability

for all

- Given a switched system $\dot{x} = f_\sigma(x)$
  - **universal**, asymptotic stability: → it will always be stable. No matter how we switch
    $$x \to 0,\ \forall\sigma$$
  - **existential**, asymptotic stability:
    $$\exists\sigma \text{ s.t. } x \to 0$$
    → there is a signal that will make it stable
- If the switch signal is generated by an underlying hybrid automaton:
  - **hybrid**, asymptotic stability:
    $$x \to 0$$

Some Results

If all the individual modes are A.S., then

Existentially A.S.

- *Why?* Simply pick a mode and never switch!

Not always universally A.S.

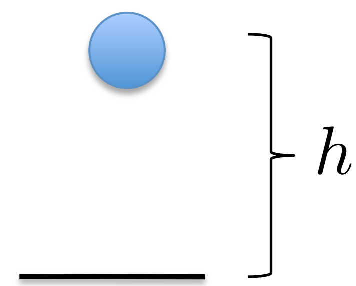
- *Why?* See the counter example
- *What to do?* (common Lyapunov function)

Practically Speaking

- Design stabilizing controllers for the subsystems
- Design the switching logic in the HA
- (Find common Lyapunov function)
- Be aware of the potential danger here and test, test, test!

Lecture 5.5 – The Bouncing Ball

- Let's model a ball bouncing on a surface:



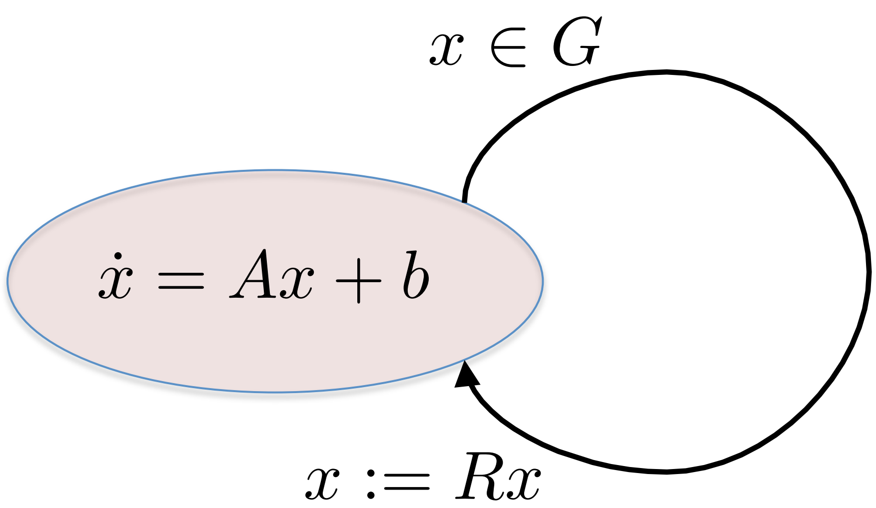
- Equations of motion in-between bounces:

$$\ddot{h} = -g \;\Rightarrow\; \dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -g \end{bmatrix}, \quad y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

- Bounces:

$$\dot{h} := -\gamma \dot{h} \;\Rightarrow\; x := \begin{bmatrix} 1 & 0 \\ 0 & -\gamma \end{bmatrix} x$$

The Ball HA



Guards?

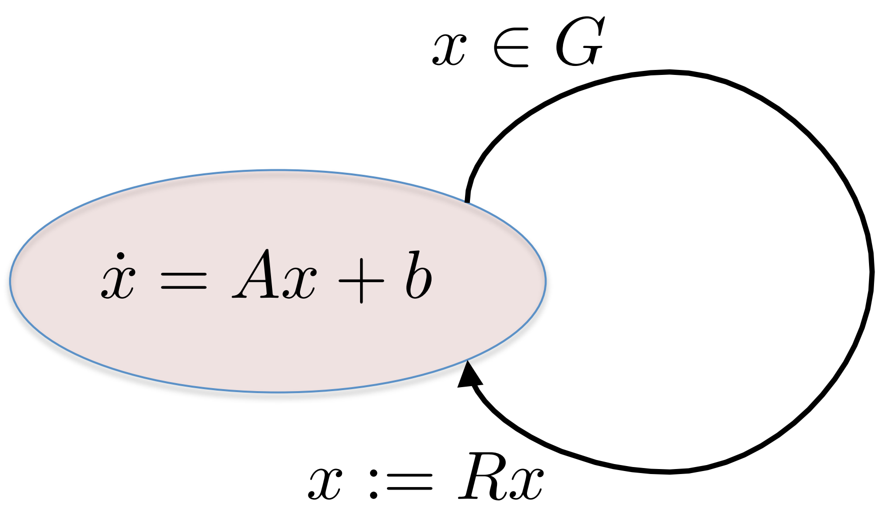
$$\cancel{h \leq 0 \;\Rightarrow\; \begin{bmatrix} 1 & 0 \end{bmatrix} x \leq 0}$$

$$h \leq 0 \text{ and } \dot{h} \leq 0 \;\Rightarrow\; x \leq 0$$

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, \quad b = \begin{bmatrix} 0 \\ -g \end{bmatrix}$$

$$R = \begin{bmatrix} 1 & 0 \\ 0 & -\gamma \end{bmatrix}$$

The Ball HA



- Is this any good?
- To answer that we need to figure out how the system actually behaves!

Solving for the Output

$$\ddot{h} = -g \;\Rightarrow\; \dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -g \end{bmatrix}, \quad y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

$$y = C\Phi(t, t_0)x(t_0) + C\int_{t_0}^{t} \Phi(t,\tau)Bu(\tau)d\tau$$

$$\Phi(t,\tau) = e^{A(t-\tau)}$$

$$A^2 = A^3 = \ldots = 0$$

$$e^{At} = \sum_{k=0}^{\infty} \frac{At^k}{k!} = I + At + 0 = \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix}$$

Solving for the Output

$$\ddot{h} = -g \;\Rightarrow\; \dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -g \end{bmatrix}, \quad y = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

$$y = C\Phi(t,t_0)x(t_0) + C\int_{t_0}^{t} \Phi(t,\tau)Bu(\tau)d\tau$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & t-t_0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} h_0 \\ \dot{h}_0 \end{bmatrix}$$

$$+ \begin{bmatrix} 1 & 0 \end{bmatrix} \int_{t_0}^{t} \begin{bmatrix} 1 & t-\tau \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ -g \end{bmatrix} d\tau$$

$$= h_0 + \dot{h}_0(t-t_0) - \frac{g}{2}(t-t_0)^2$$

Time In-Between Bounces?

$$y = h_0 + \dot{h}_0(t - t_0) - \frac{g}{2}(t - t_0)^2$$

$$h_0 = 0, \quad t_0 = 0$$

$$y(T) = 0 = \dot{h}_0 T - \frac{g}{2}T^2 = T(\dot{h}_0 - g/2T)$$

$$T = 0, \quad T = \frac{2\dot{h}_0}{g}$$

Accumulated Bounce Times

$$T = \frac{2\dot{h}_0}{g}$$ (time in-between bounces)

velocity at the beginning

$$\dot{h}_{0,0} = v$$

time of first bounce

$$T_1 = \frac{2v}{g}$$

velocity after first bounce

$$\dot{h}_{0,1} = \gamma v$$

time of second bounce

$$T_2 = \frac{2v}{g} + \gamma\frac{2v}{g}$$

$$\vdots$$

$$\dot{h}_{0,k} = \gamma^k v$$

$$T_N = \frac{2v}{g}\sum_{k=0}^{N-1}\gamma^k \rightarrow \frac{2v}{g}\frac{1}{1-\gamma}$$

$$\gamma < 1$$

So What?

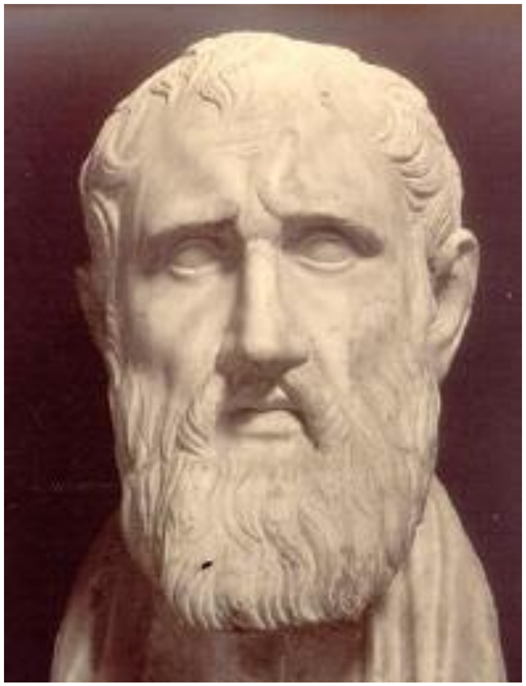
$$T_\infty = \frac{2v}{g}\frac{1}{1-\gamma} \quad < \infty$$

- The ball bounces an infinite number of times in finite time.
- More than just a mathematical curiosity
  - Simulations crash!
  - Hybrid system is undefined beyond this time!
- This is known as the ***Zeno Phenomenon*** – next lecture

Lecture 5.6 – The Zeno Phenomenon

- Problem with the bouncing ball: Infinitely many switches in finite time
- This is bad:
  - Simulations crash
  - Model is not accurate
  - System behavior fundamentally ill-defined beyond "Zeno point"

The Zeno Phenomenon

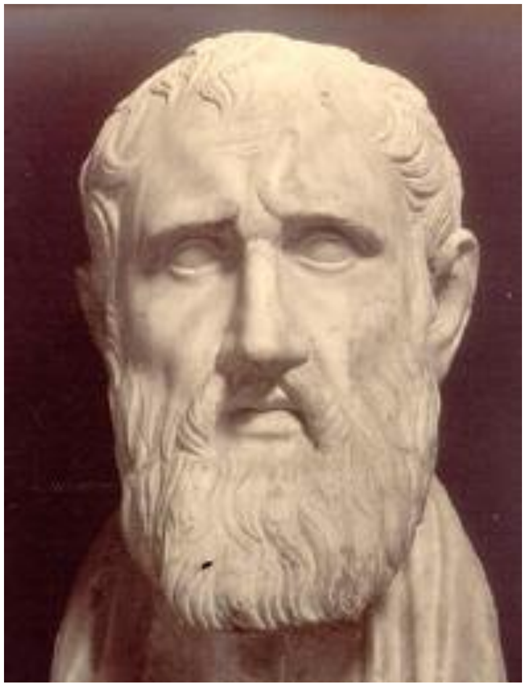


Zeno of Elea: 490 BC-430 BC

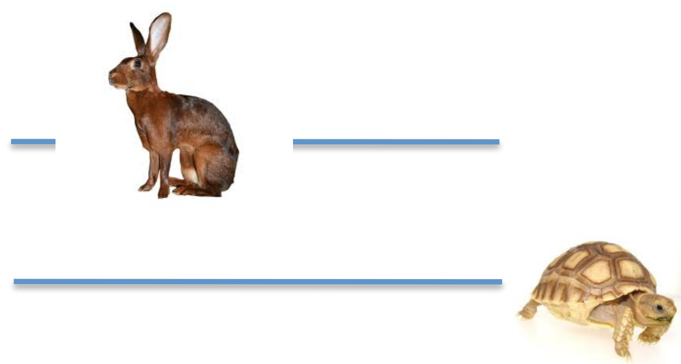
The hare

The tortoise

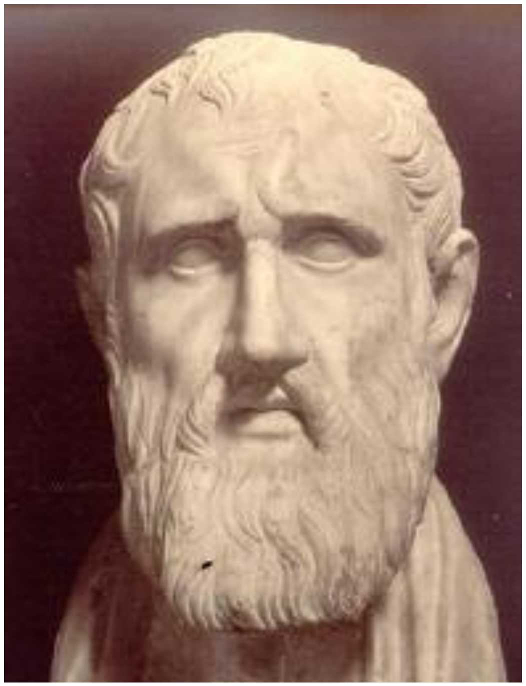
The Zeno Phenomenon



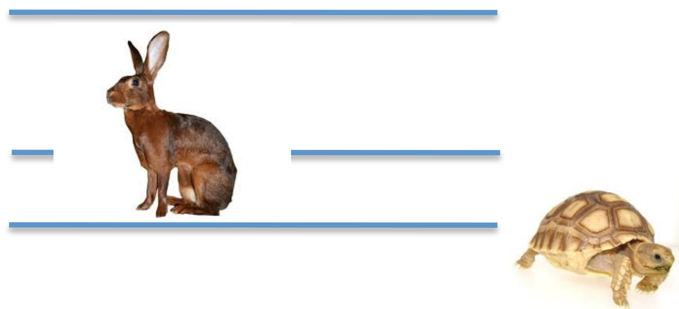
Zeno of Elea: 490 BC-430 BC



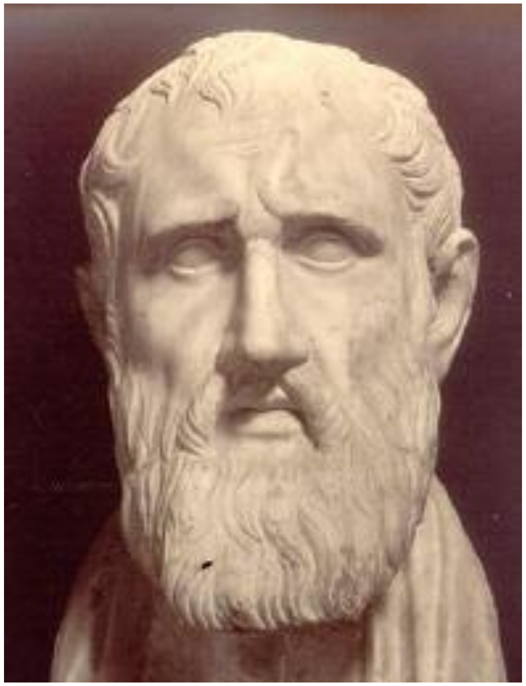
The Zeno Phenomenon



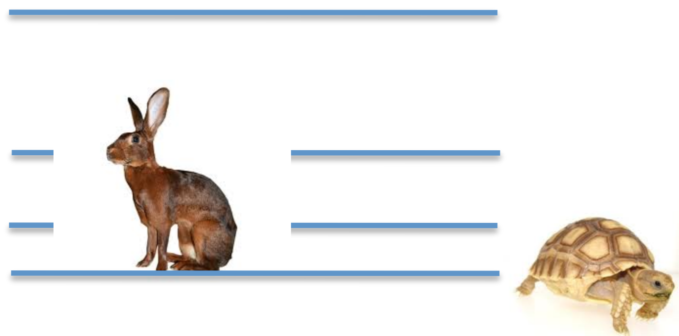
Zeno of Elea: 490 BC-430 BC



The Zeno Phenomenon



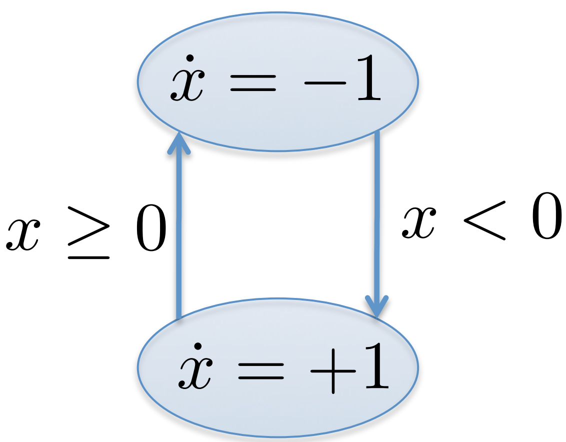
Zeno of Elea: 490 BC-430 BC



The Paradox: The hare never catches up with the tortoise
The Solution: Convergent series
The Problem: Infinitely many switches in finite time

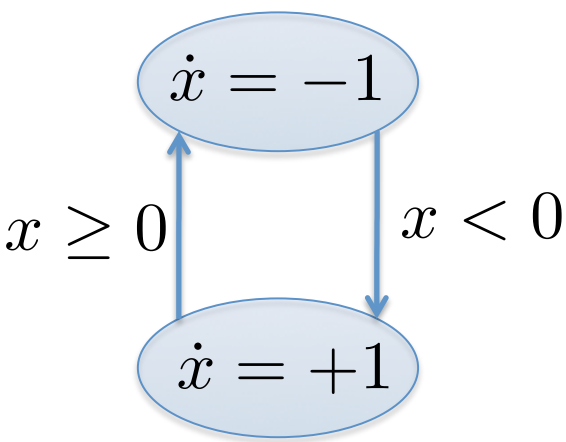
Example

$$\dot{x} = \begin{cases} -1 & x \geq 0 \\ +1 & x < 0 \end{cases}$$



Super-Zeno?

$$\dot{x} = \begin{cases} -1 & x \geq 0 \\ +1 & x < 0 \end{cases}$$

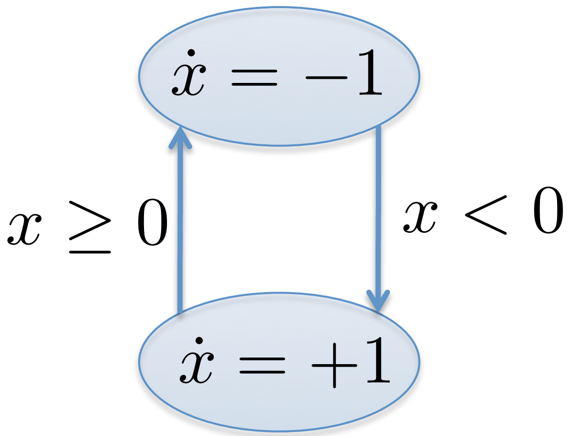


- This system switches infinitely many times in a single time-instant
- Zeno: Infinitely many switches in finite time
  - Type 1: In a single time-instant
  - Type 2: Not Type 1 (bouncing ball)

Good News and Bad News

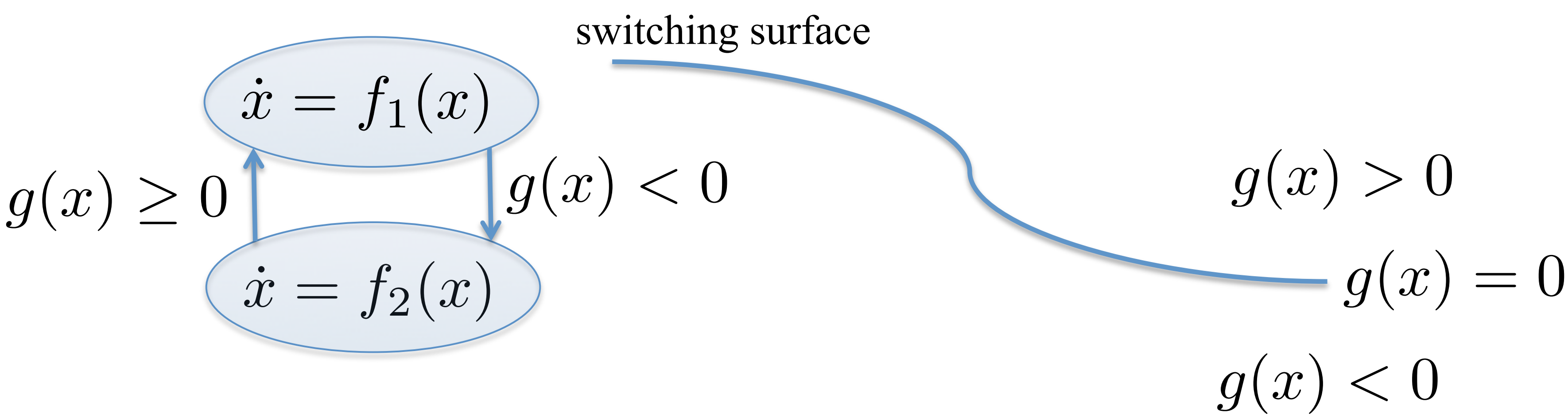
- Zeno is a problem
- Type 1 is not only detectable, but one can deal with it in a rather straightforward manner
- Type 2 is overall hard to handle!

Lecture 5.7 – Sliding Mode Control



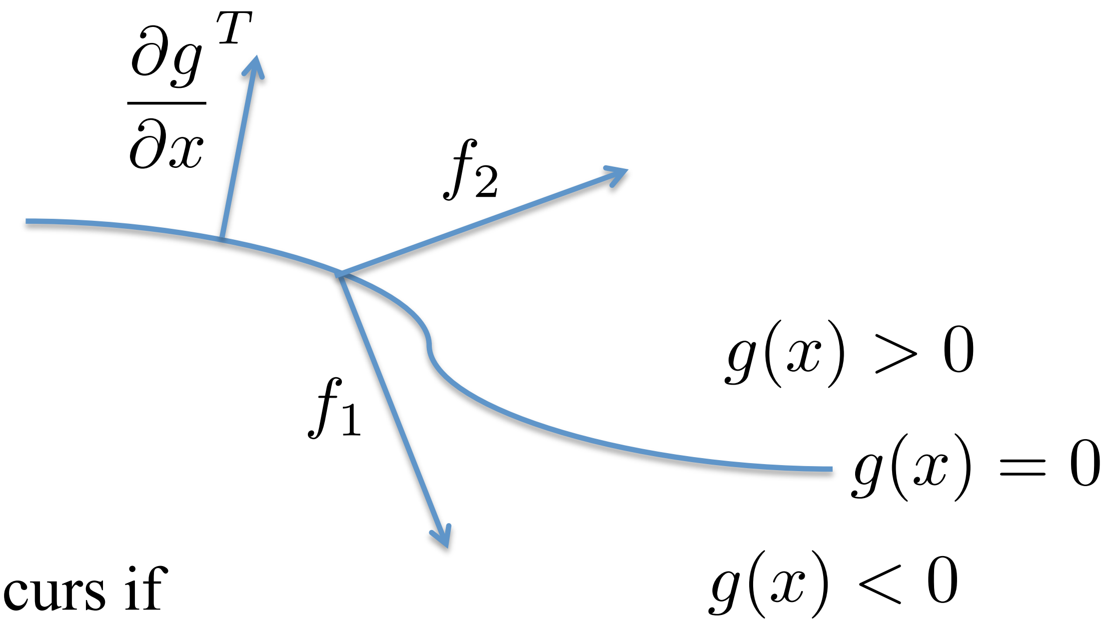
- It is clear what should happen!
- How do we make that mathematically sound?
- Sliding Mode Control

Switching Surfaces



- Both vector fields point “inwards” = bad!
- We should keep sliding along the switching surface
- Sliding Mode Control

Sliding?

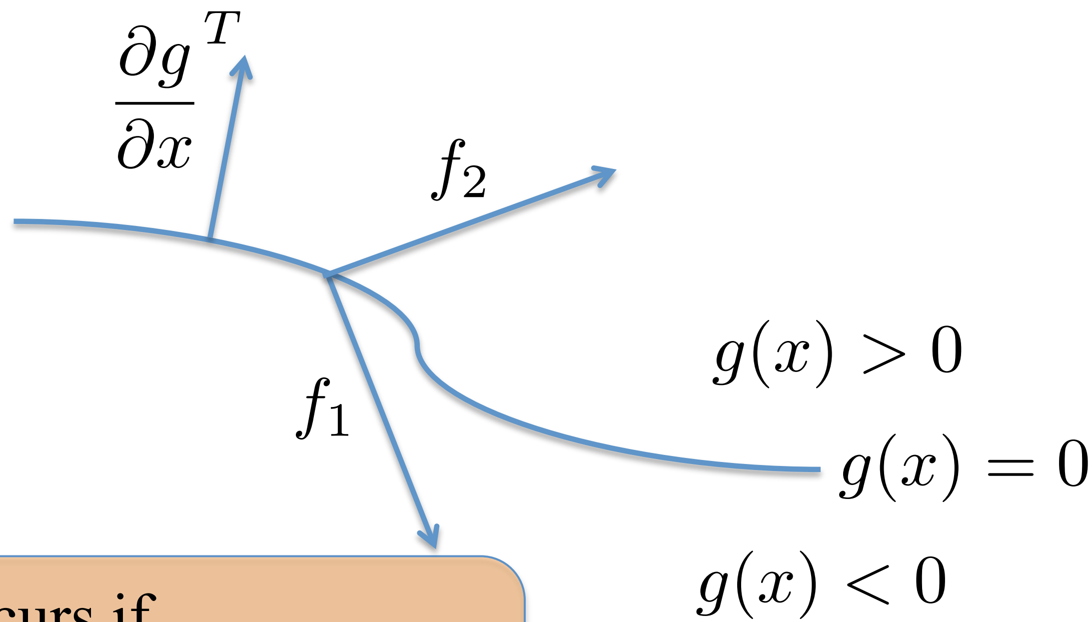


Sliding occurs if

$$\frac{\partial g}{\partial x} f_1 < 0 \text{ AND } \frac{\partial g}{\partial x} f_2 > 0$$

derivative of g in direction f = $L_f g$ = Lie derivative

Sliding?



Sliding occurs if

$$L_{f_1} g < 0 \text{ AND } L_{f_2} g > 0$$

derivative of g in direction f = $L_f g$ = Lie derivative

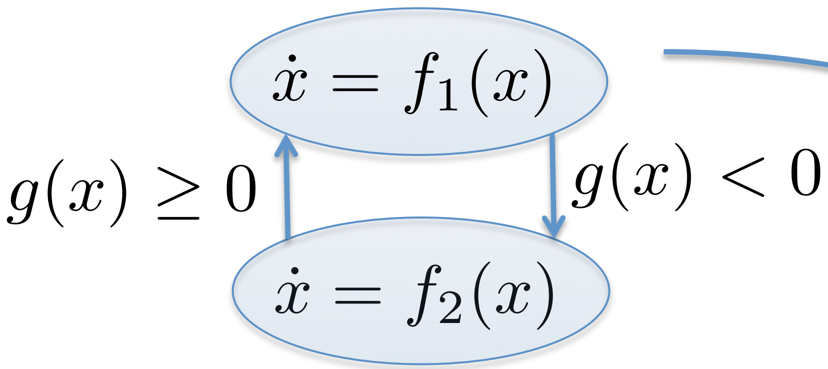
A Test For Type 1 Zeno

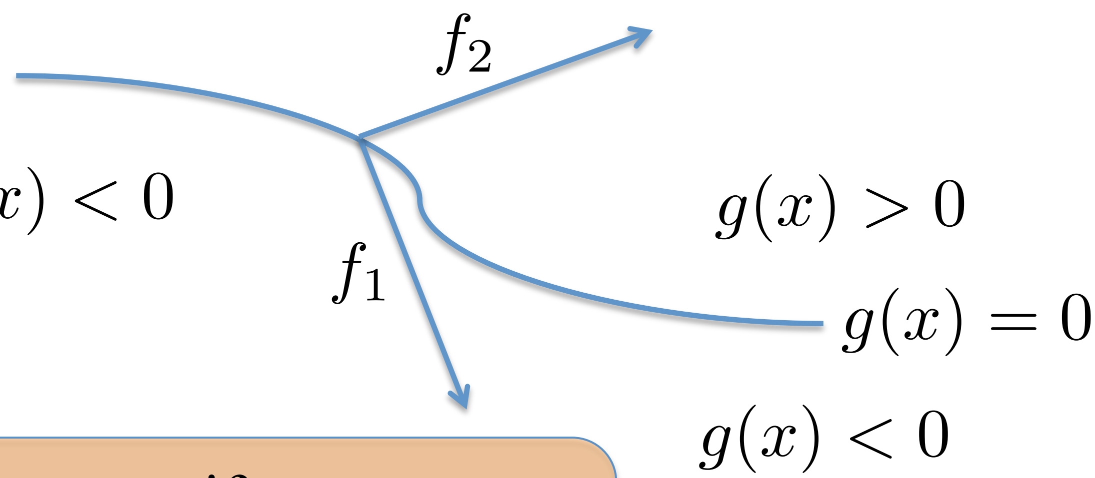
Sliding occurs if

$$L_{f_1} g < 0 \text{ AND } L_{f_2} g > 0$$

- Next time: But what happens beyond the Zeno point?

Lecture 5.8 – Regularizations



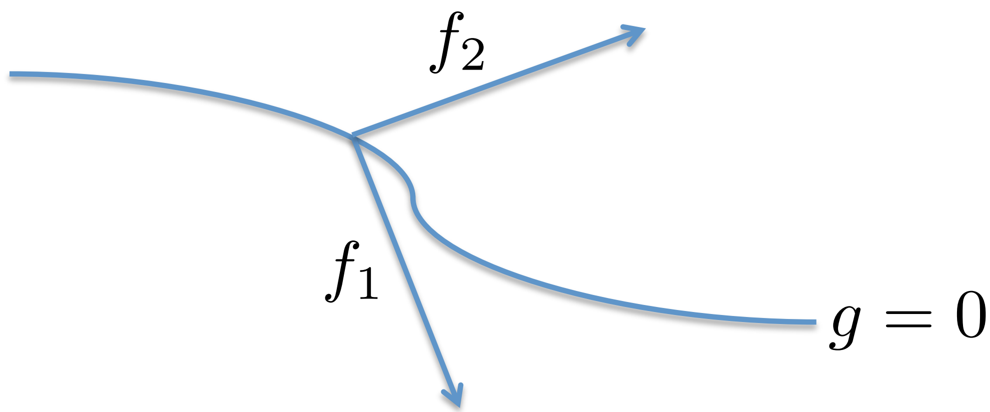


Sliding occurs if

$$L_{f_1} g < 0 \text{ AND } L_{f_2} g > 0$$

- *How do we move beyond the Zeno point?*

The Sliding "Mode"



$$\frac{dg}{dt} = 0$$

$$\frac{dg}{dt} = \frac{\partial g}{\partial x}\dot{x} = \frac{\partial g}{\partial x}(\sigma_1 f_1 + \sigma_2 f_2) = \sigma_1 L_{f_1} g + \sigma_2 L_{f_2} g$$

$$= 0 \;\Rightarrow\; \sigma_2 = -\sigma_1 \frac{L_{f_1} g}{L_{f_2} g}$$

$$\sigma_1, \sigma_2 \geq 0,\; \sigma_1 + \sigma_2 = 1$$

Back to the Example

$$\dot{x} = \begin{cases} -1 & x \geq 0 \\ +1 & x < 0 \end{cases} \qquad g(x) = x = 0$$

$$L_{f_1} g = \frac{\partial g}{\partial x} f_1 = 1(-1) = -1$$

$$L_{f_1} g = \frac{\partial g}{\partial x} f_2 = 1 \cdot 1 = 1$$

$$\sigma_2 = -\sigma_1 \frac{L_{f_1} g}{L_{f_2} g} = -\sigma_1 \frac{-1}{1} = \sigma_1$$

$$\sigma_1 = \sigma_2 = 0.5$$

The Induced Mode

$$\dot{x} = 0.5 f_1 + 0.5 f_2 = 0$$

The Induced Mode

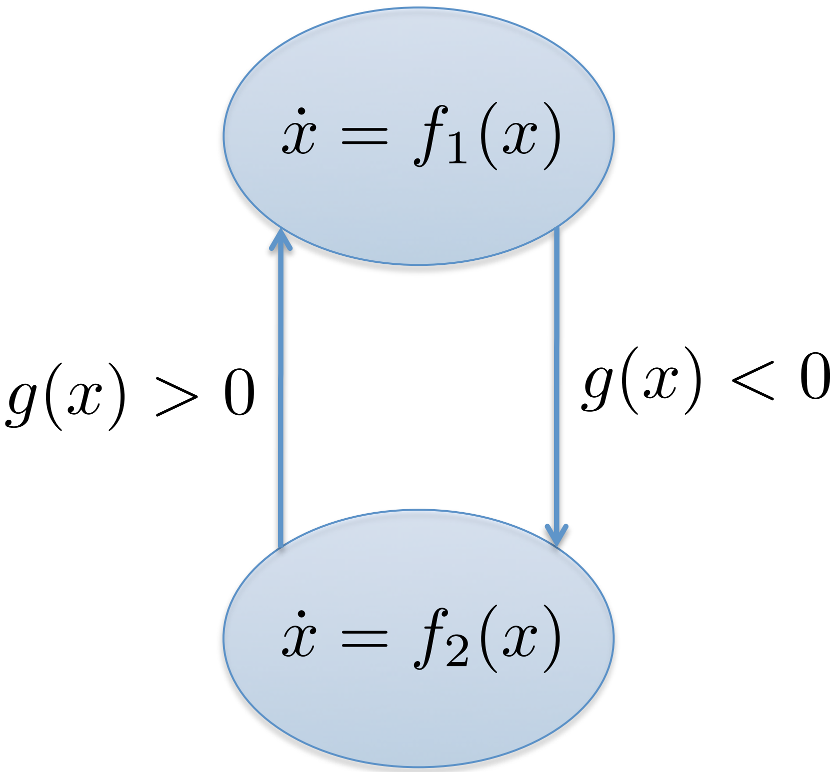
$$\sigma_2 = -\sigma_1 \frac{L_{f_1} g}{L_{f_2} g} \qquad \sigma_1 + \sigma_2 = \sigma_1 \left(1 - \frac{L_{f_1} g}{L_{f_2} g}\right) = 1$$

$$\sigma_1 = \frac{1}{1 - \frac{L_{f_1} g}{L_{f_2} g}} = \frac{L_{f_2} g}{L_{f_2} g - L_{f_1} g}$$

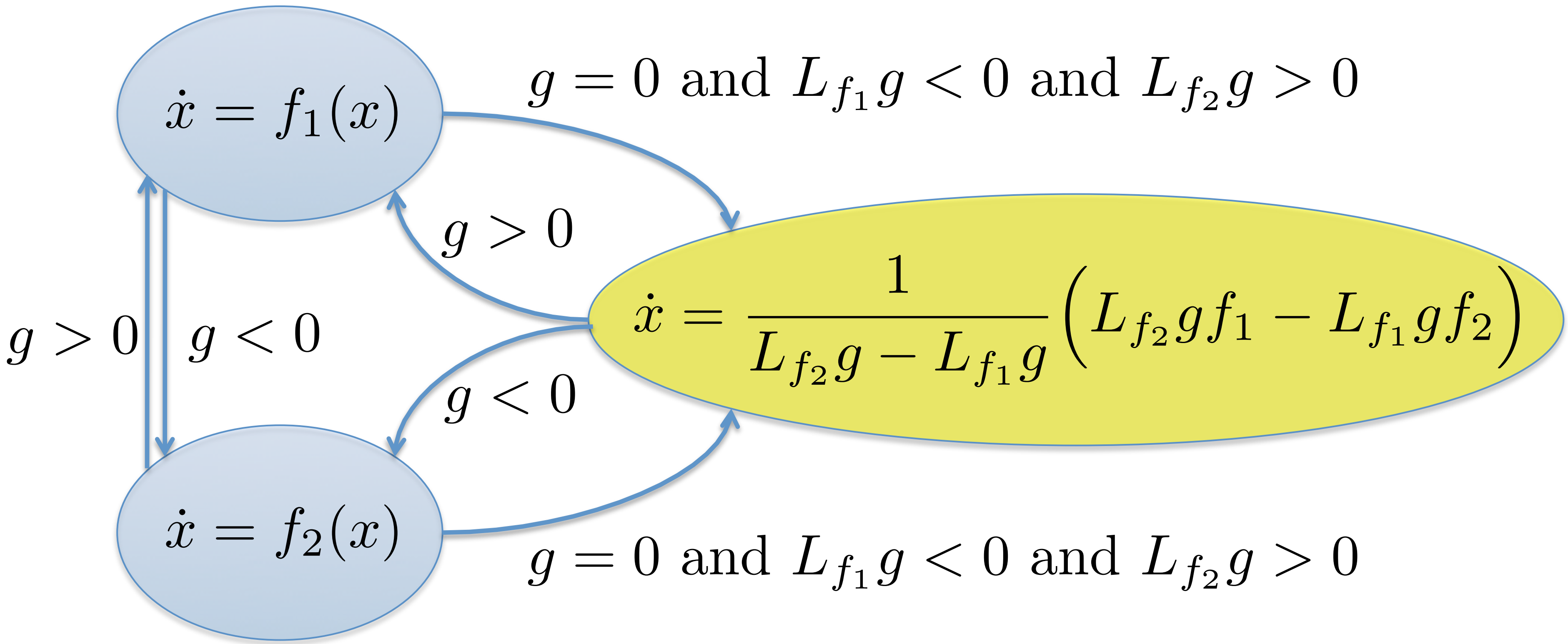
$$\sigma_2 = -\frac{L_{f_1} g}{L_{f_2} g - L_{f_1} g}$$

$$\dot{x} = \frac{1}{L_{f_2} g - L_{f_1} g} \left(L_{f_2} g f_1 - L_{f_1} g f_2\right)$$

Regularizations of Type 1 Zeno HA



Regularizations of Type 1 Zeno HA



Hybrid Systems: In Summary

- We have
  - Models
  - Stability Awareness
  - Zeno
  - Regularizations
- Next Module: Back to **ROBOTICS**!