

Lecture 17: Modeling Cyberphysical Systems

Hybrid systems

Huan Zhang

huan@huan-zhang.com

Deadlines

Homework 3 will be released this week

Deadline: 2 weeks after homework posted

Review: dynamical systems

Behaviors of physical processes are described in terms of instantaneous laws

Common notation: $\frac{dx(t)}{dt} = f(x(t), u(t), t) - Eq. (1)$

where time $t \in \mathbb{R}$; **state** $x(t) \in \mathbb{R}^n$; **input** $u(t) \in \mathbb{R}^m$; $f: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}^n$

Example. $\frac{dx(t)}{dt} = v(t)$; $\frac{dv(t)}{dt} = -g$

Initial value problem: Given system (1) and initial state $x_0 \in \mathbb{R}^n$, $t_0 \in \mathbb{R}$, and input $u: \mathbb{R} \rightarrow \mathbb{R}^m$, find a state trajectory or *solution* ξ of (1).

Review: Linear time invariant system

$$\dot{x}(t) = Ax(t) + Bu(t)$$

Define Matrix exponential:

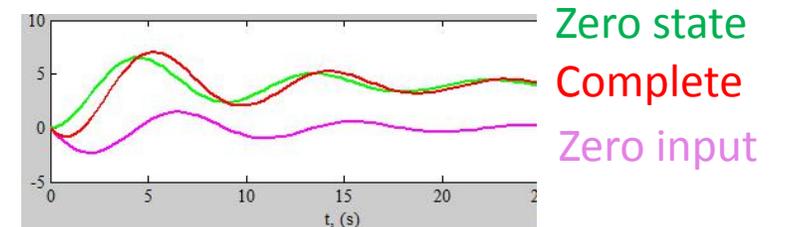
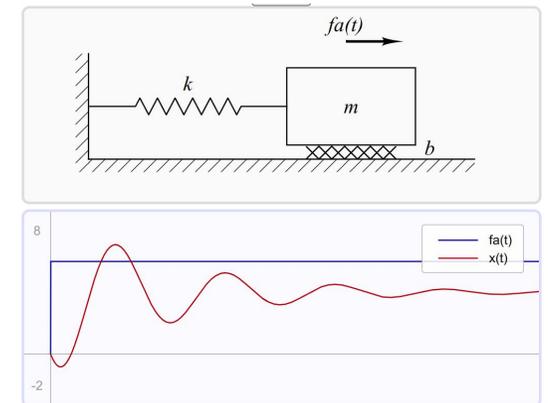
$$e^{At} = 1 + At + \frac{1}{2!}(At)^2 + \dots = \sum_0^{\infty} \frac{1}{k!}(At)^k$$

Theorem. $\xi(t, x_0, u) = \Phi(t)x_0 + \int_{t_0}^t e^{A(t-\tau)}Bu(\tau)d\tau$

Zero input

Zero state

Here $\Phi(t) := e^{At}$ is the state-transition matrix



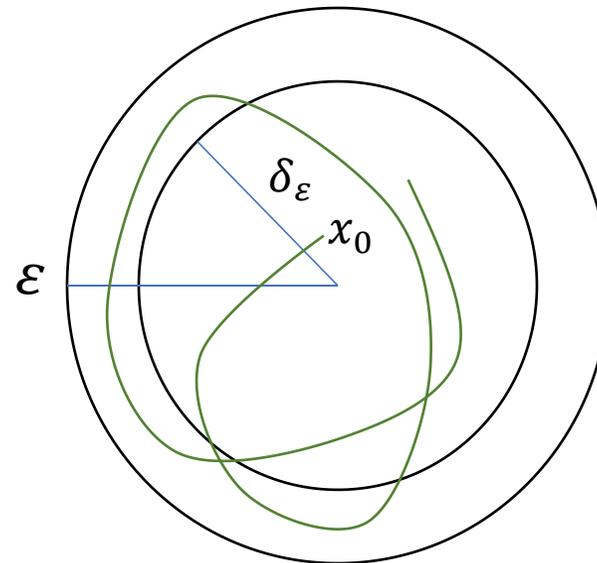
Review Lyapunov stability

Lyapunov stability: The system $\dot{x}(t) = f(x(t))$ is said to be **Lyapunov stable** (at the origin) if

$$\forall \varepsilon > 0, \exists \delta_\varepsilon > 0 \text{ such that } |x_0| \leq \delta_\varepsilon \Rightarrow \forall t \geq 0, |\xi(x_0, t)| \leq \varepsilon.$$

“if we start the system close enough to the equilibrium, it remains close enough”

How is this related to invariants and reachable states ?

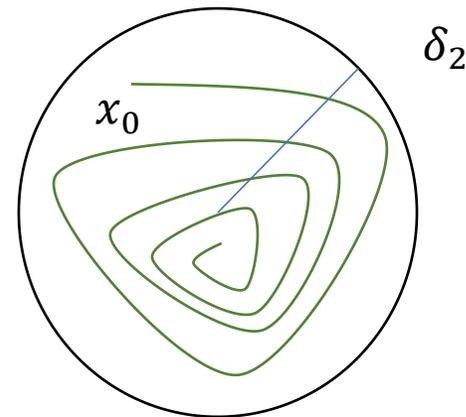


Review: Asymptotically stability

The system $\dot{x}(t) = f(x(t))$ is said to be ***Asymptotically stable*** (at the origin) if it is Lyapunov stable and

$\exists \delta_2 > 0$ such that $\forall |x_0| \leq \delta_2$ as $t \rightarrow \infty$, $|\xi(x_0, t)| \rightarrow \mathbf{0}$.

If the property holds for any δ_2 then **Globally Asymptotically Stable**



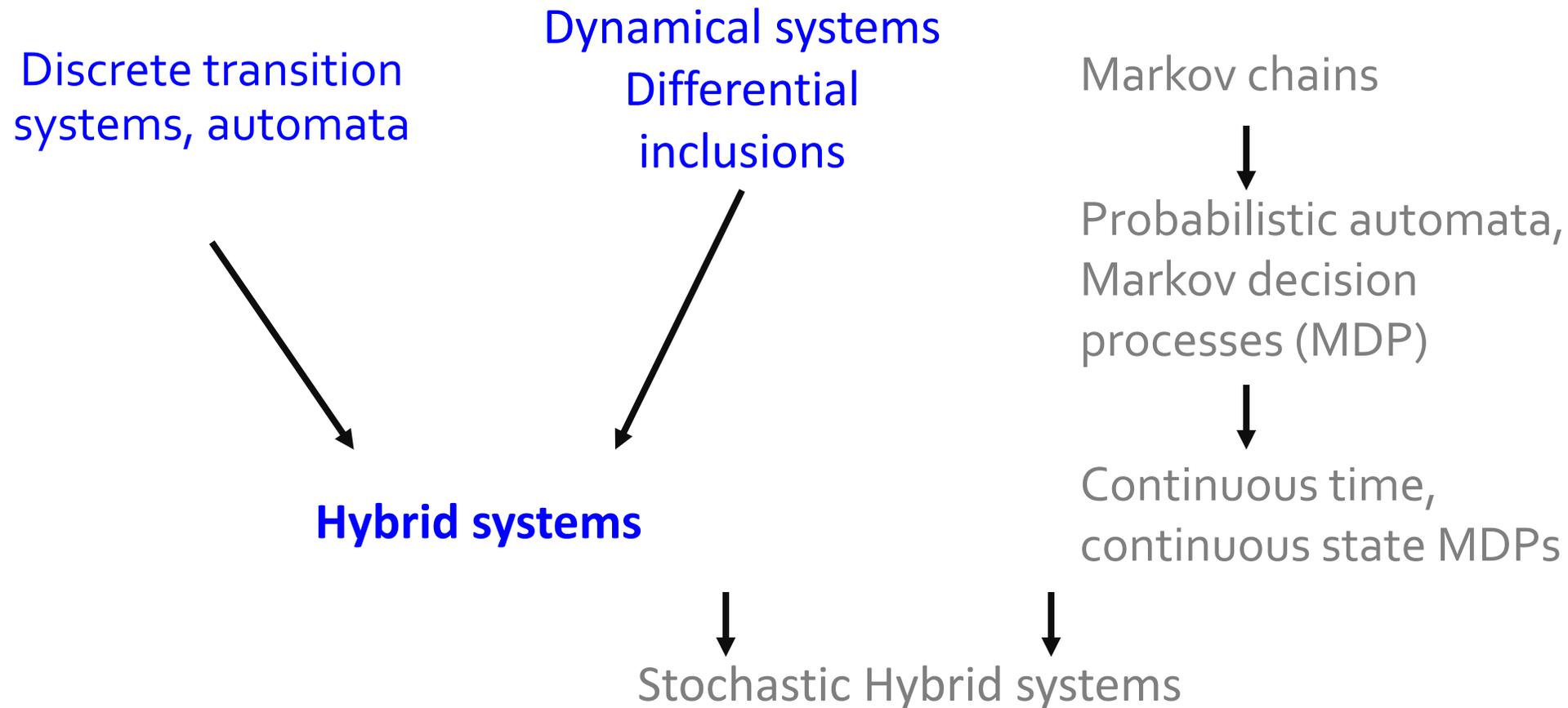
Review: Verifying Stability

Theorem. (Lyapunov) Consider the system (1) with state space $x \in \mathbb{R}^n$ and suppose there exists a positive definite, continuously differentiable function $V: \mathbb{R}^n \rightarrow \mathbb{R}$. The system is:

1. Lyapunov stable if $\dot{V}(x) = \frac{\partial V}{\partial x} f(x) \leq 0$, for all $x \neq 0$
2. Asymptotically stable if $\dot{V}(x) < 0$, for all $x \neq 0$
3. It is globally AS if V is also radially unbounded.

(V is radially unbounded if $\|x\| \rightarrow \infty \Rightarrow V(x) \rightarrow \infty$)

Today's lecture: Hybrid systems



Outline

- Hybrid automata
- Executions
- Special kinds of executions: admissible, Zeno
- Hybrid stability

Recall from Lecture 2. language defines an automaton

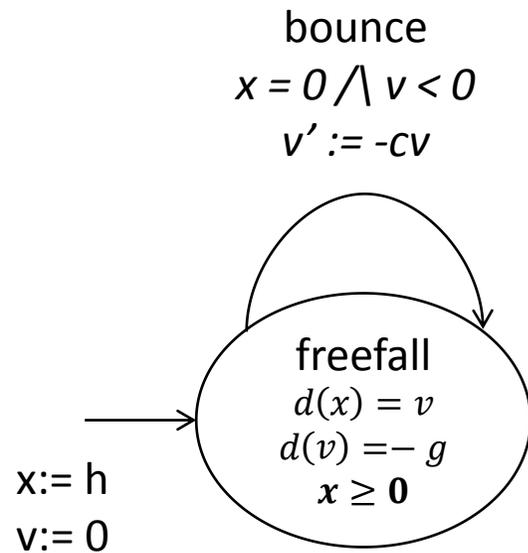
An **automaton** is a tuple $\mathcal{A} = \langle X, \Theta, A, \mathcal{D} \rangle$ where

- X is a set of names of variables; each variable $x \in X$ is associated with a type, $type(x)$
 - A **valuation** for X maps each variable in X to its type
 - Set of all valuations: $val(X)$ this is sometimes identified as the **state space** of the automaton
- $\Theta \subseteq val(X)$ is the set of **initial** or **start states**
- A is a set of names of **actions** or **labels**
- $\mathcal{D} \subseteq val(X) \times A \times val(X)$ is the set of **transitions**
 - a transition is a triple (u, a, u')
 - We write it as $u \rightarrow_a u'$

```
automaton DijkstraTR(N:Nat, K:Nat), where K > N
type ID: enumeration [0,...,N-1]
type Val: enumeration [0,...,K]
actions
  update(i:ID)
variables
  x:[ID -> Val]
transitions
  update(i:ID)
    pre i = 0  $\wedge$  x[i] = x[N-1]
    eff x[i] := (x[i] + 1) % K

  update(i:ID)
    pre i > 0  $\wedge$  x[i]  $\sim$  x[i-1]
    eff x[i] := x[i-1]
```

Bouncing Ball: Hello world of CPS



Graphical Representation used in many articles

automaton Bouncingball(c,h,g)

variables: x : Reals := h, v : Reals := 0

actions: bounce

transitions:

bounce

pre $x = 0 \wedge v < 0$

eff $v := -cv$

trajectories:

freefall

evolve $d(x) = v; d(v) = -g$

invariant $x \geq 0$



mode invariant,
not to be
confused with
invariants of the
automaton

Trajectories

Given a set of variables X and a **time interval** J which can be of the form $[0, T]$, $[0, T)$ or $[0, \infty)$, a **trajectory** for X is a function $\tau: J \rightarrow \text{val}(X)$

We will specify τ as **solutions of differential equations**

The **first state** of a trajectory τ . $fstate := \tau(0)$

If τ is right closed then the **limit state** of a trajectory τ . $lstate = \tau(T)$

If τ is finite then **duration** of τ is $\tau.dur = T$

Trajectories

Given a set of variables X and a **time interval** J which can be of the form $[0, T]$, $[0, T)$ or $[0, \infty)$, a **trajectory** for X is a function $\tau: J \rightarrow \text{val}(X)$

The domain of τ . $\text{dom} = J$

A **point trajectory** is a trajectory with $\tau.\text{dom} = [0,0]$

Operations on trajectories: prefix, suffix, concatenation

A **prefix** τ' of a trajectory $\tau: [0, T] \rightarrow \text{val}(X)$, is a function $\tau': [0, T'] \rightarrow \text{val}(X)$ such that $T' \leq T$ and $\tau'(t) = \tau(t)$ for all $t \in \tau'.\text{dom}$

Hybrid Automaton

$\mathcal{A} = (X, \Theta, A, \mathcal{D}, \mathcal{T})$

- X : set of **state variables**
 - $Q \subseteq \text{val}(X)$ set of **states**
- $\Theta \subseteq Q$ set of **start states**
- set of **actions** A
- Set of **transitions** $\mathcal{D} \subseteq Q \times A \times Q$
- \mathcal{T} : set of **trajectories**

automaton Bouncingball(c,h,g)

variables: x : Reals := h , v : Reals := 0

actions: bounce

transitions:

bounce

pre $x = 0 \wedge v < 0$

eff $v := -cv$

trajectories:

freefall

evolve $d(x) = v$; $d(v) = -g$

invariant $x \geq 0$

Semantics: Executions and Traces

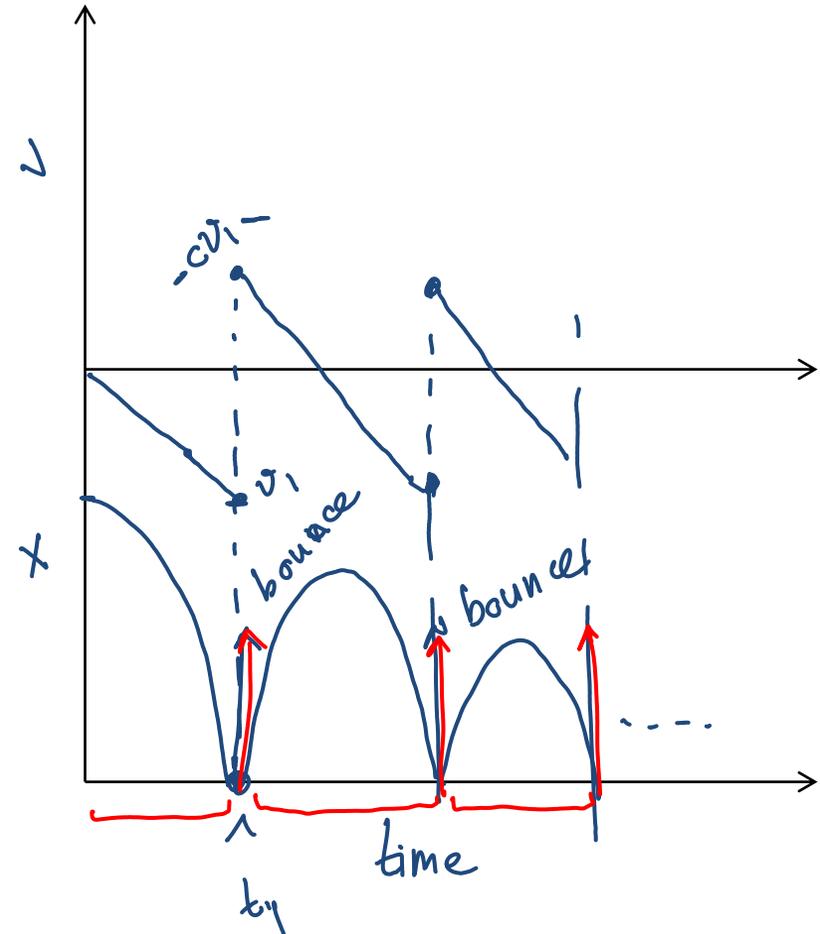
- An **execution fragment** of \mathcal{A} is an (possibly infinite) alternating (A, X)-sequence $\alpha = \tau_0 a_1 \tau_1 a_2 \tau_2 \dots$ where
 - $\forall i, \tau_i.lstate \xrightarrow{a_{i+1}} \tau_{i+1}.fstate$
- If $\tau_0.fstate \in \Theta$ then α is an **execution**
- **Execs** $_{\mathcal{A}}$ set of all executions
- The first state of an execution α is:
 $\alpha.fstate = \tau_0.fstate$
- If the execution α is **finite and closed**:
 $\tau_0 a_1 \tau_1 a_2 \tau_2 \dots \tau_k$ then $\alpha.lstate = \tau_k.lstate$
- A state x is reachable if there exists an execution α with $\alpha.lstate = x$

Semantics: Executions and Traces

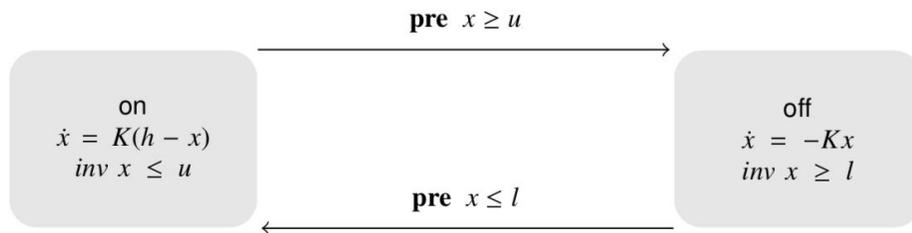
- An **execution fragment** of \mathcal{A} is an (possibly infinite) alternating (A, X)-sequence $\alpha = \tau_0 a_1 \tau_1 a_2 \tau_2 \dots$ where
 - $\forall i, \tau_i.lstate \xrightarrow{a_{i+1}} \tau_{i+1}.fstate$
- If $\tau_0.fstate \in \Theta$ then α is an **execution**
- **Execs** $_{\mathcal{A}}$ set of all executions
- The first state of an execution α is:

$$\alpha.fstate = \tau_0.fstate$$
- If the execution α is **finite and closed**:

$$\tau_0 a_1 \tau_1 a_2 \tau_2 \dots \tau_k$$
 then $\alpha.lstate = \tau_k.lstate$
- A state x is **reachable** if there exists an execution α with $\alpha.lstate = x$



Example: Thermostat



automaton Thermostat($u, l, K, h : \text{Real}$) where $u > l$

type *Status* enumeration [*on*, *off*]

actions

turnOn; turnOff;

variables

$x : \text{Real} := l$ $loc : \text{Status} := on$

transitions

turnOn

pre $x \leq l \wedge loc = off$

eff $loc := on$

turnOff

pre $x \geq u \wedge loc = on$

eff $loc := off$

trajectories

modeOn

evolve $d(x) = K(h - x)$

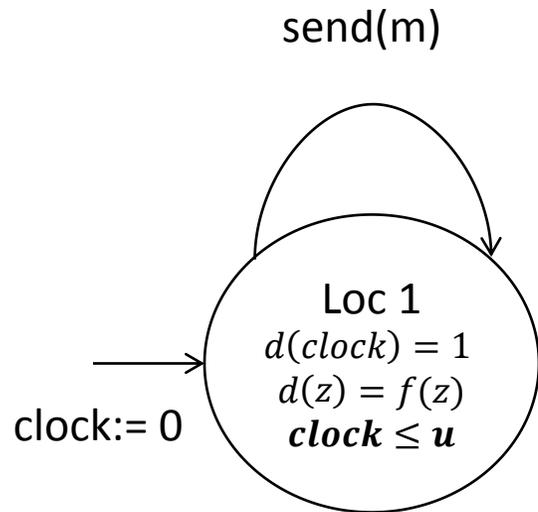
invariant $loc = on \wedge x \leq u$

modeOff

evolve $d(x) = -Kx$

invariant $loc = off \wedge x \geq l$

Another Example: Periodically Sending Process



Automaton PeriodicSend(u, f)

variables:

$clock: \text{Reals} := 0, z: \text{Reals}$

actions: $send(m: \text{Reals})$

transitions:

$send(m)$

pre $(clock = u) \wedge (m = z)$

eff $clock := 0$

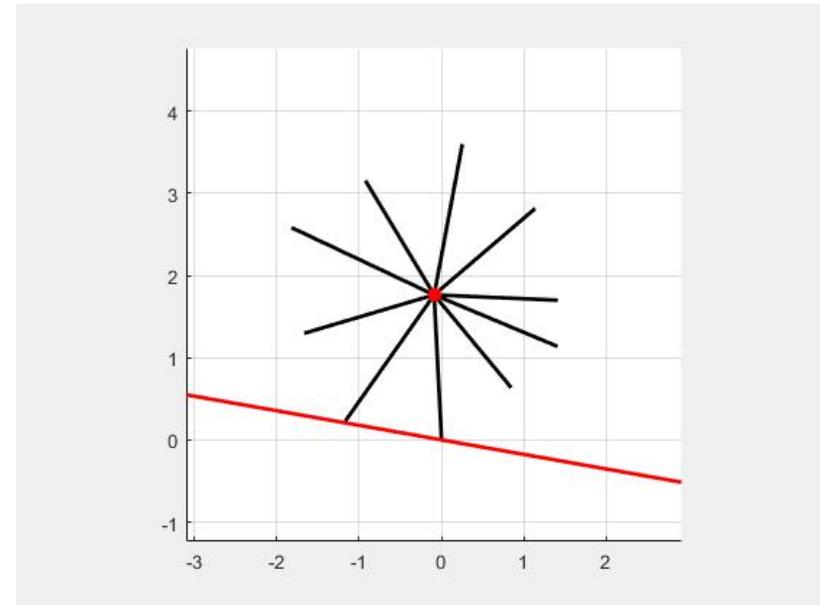
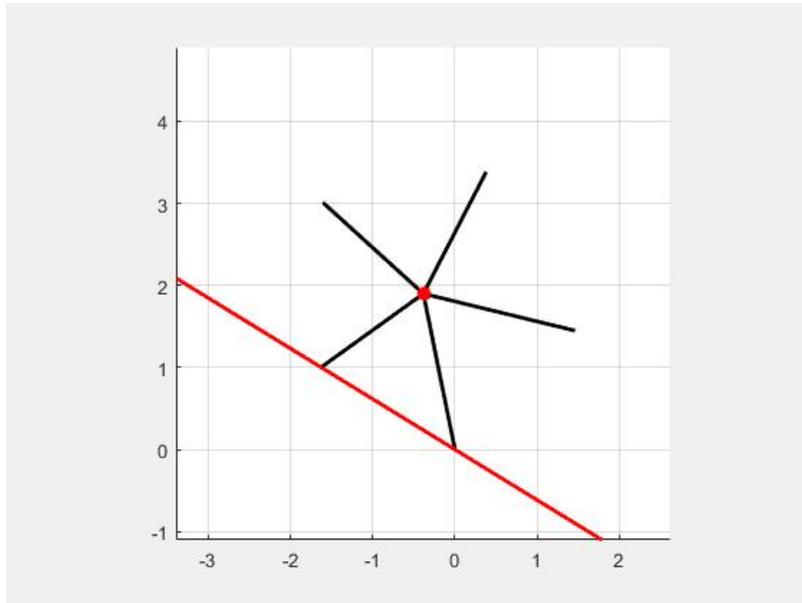
trajectories:

Loc1

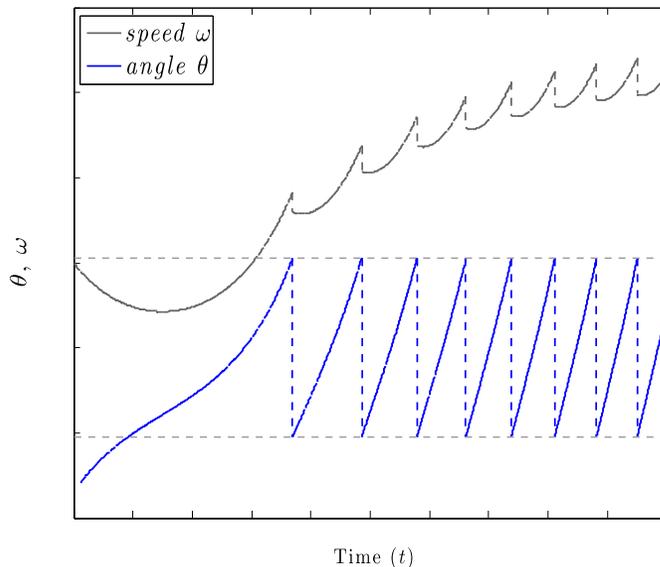
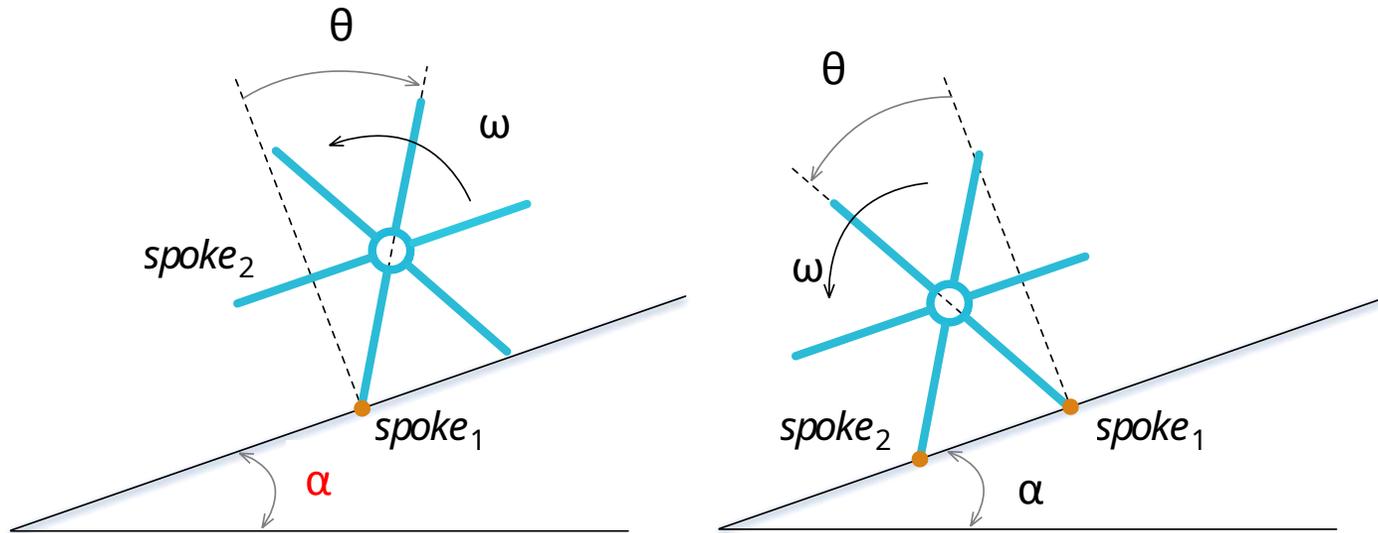
evolve $d(clock) = 1, d(z) = f(z)$

invariant $clock \leq u$

Rimless wheel: another example of hybrid system



Rimless wheel



automaton RimlessWheel(α, μ : Real, n : Nat)

const β : Real := $2 \pi / n$

type Spokes: enumeration [1,...,n]

actions

impact

variables

pivot: Spokes := 1

θ : Real := 0

ω : Real := 0

transitions

impact

pre $\theta \geq \beta / 2$

eff pivot := pivot + 1 mod n

$\theta := -\beta / 2$

$\omega := \mu \omega$

trajectories

swing

evolve

$d(\theta) = \omega$

$d(\omega) = \sin(\theta + \alpha)$

invariant $\theta \leq \frac{\beta}{2}$

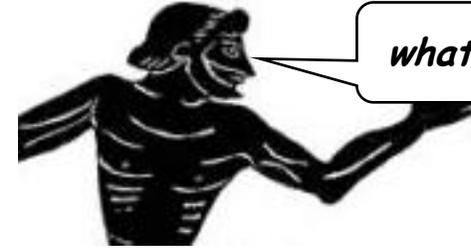
Special kinds of executions

- **Infinite:** Infinite sequence of transitions and trajectories
 $\tau_0 a_1 \tau_1 a_2 \tau_2 \dots$
- **Closed:** Finite with final trajectory with closed domain
 $\tau_0 a_1 \tau_1 a_2 \tau_2 \dots \tau_k$ and $\tau_k \cdot dom = [0, T]$
- **Admissible:** Infinite duration
 - May or may not be infinite
 - (infinite) $\tau_0 a_1 \tau_1 a_2 \tau_2 \dots$
 - (not infinite) $\tau_0 a_1 \tau_1 a_2 \tau_2 \dots \tau_k$ with $\tau_k \cdot dom = [0, \infty)$
- **Zeno:** Infinite but not admissible
 - Infinite number of transitions in finite time

Zeno's Paradox

Achilles, the fastest athlete, greatest warrior

Zeno, Greek philosopher

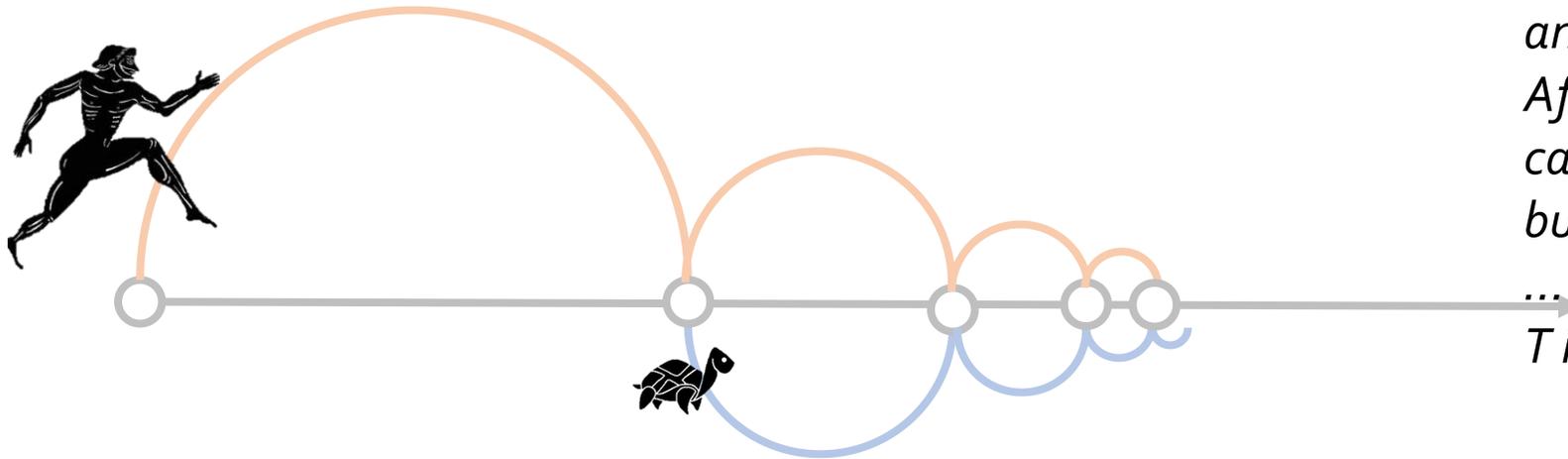


whatever!

You couldn't even beat a turtle



*Achilles runs 10 times faster than than the turtle, but the turtle gets to start 1 second earlier.
Can Achilles ever catch Turtle?*



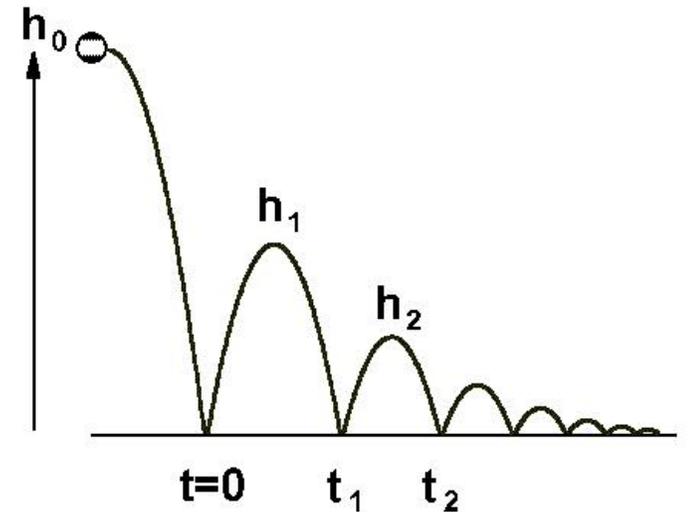
After $1/10^{\text{th}}$ of a second, Achilles reaches where the Turtle (T) started, and T has a head start of $1/10^{\text{th}}$ second. After another $1/100^{\text{th}}$ of a second, A catches up to where T was at $t=1/10$ sec, but T has a head start of $1/100^{\text{th}}$

*...
T is always ahead ...*

Lesson: Mixing discrete transitions with continuous motion can be tricky!

Zeno's Paradox: bouncing ball

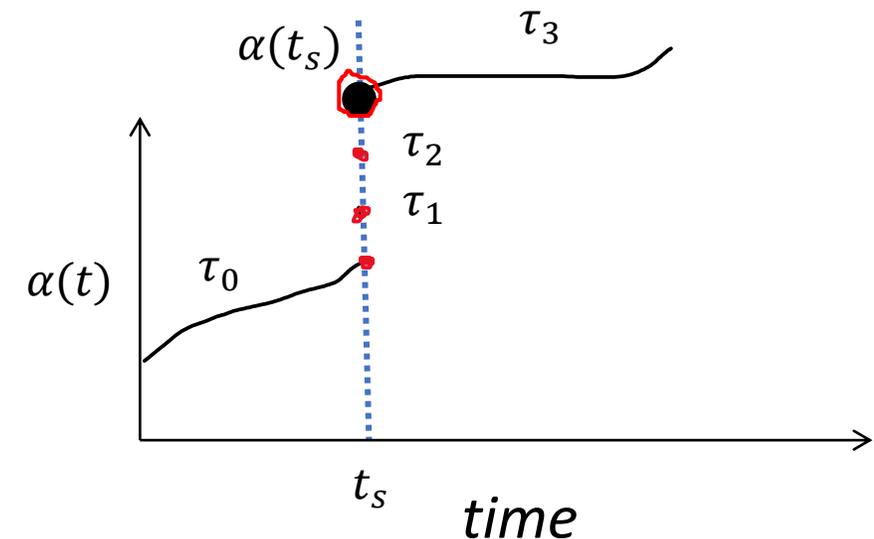
- Infinite number of bounces to come to rest!
- Yet it needs finite time to come to rest
- The time for each bounce (e.g., $t_1 - t_0$, $t_2 - t_1$, etc) is a geometric series



Read: <https://www.millersville.edu/physics/experiments/045/>

Defining stability of hybrid automata

- Given an *admissible* (infinite duration) execution $\alpha = \tau_0 a_1 \tau_1 a_2 \tau_2 \dots$
- To reason about stability of an execution, we would like to view an execution as $\alpha: [0, \infty) \rightarrow \text{val}(X)$
- But, how can we define $\alpha(t)$?
- define $\alpha(t_s) = \alpha'.lstate$ where α' is the longest **prefix** of α with $\alpha'.ltime = t_s$



Defining stability of hybrid automata

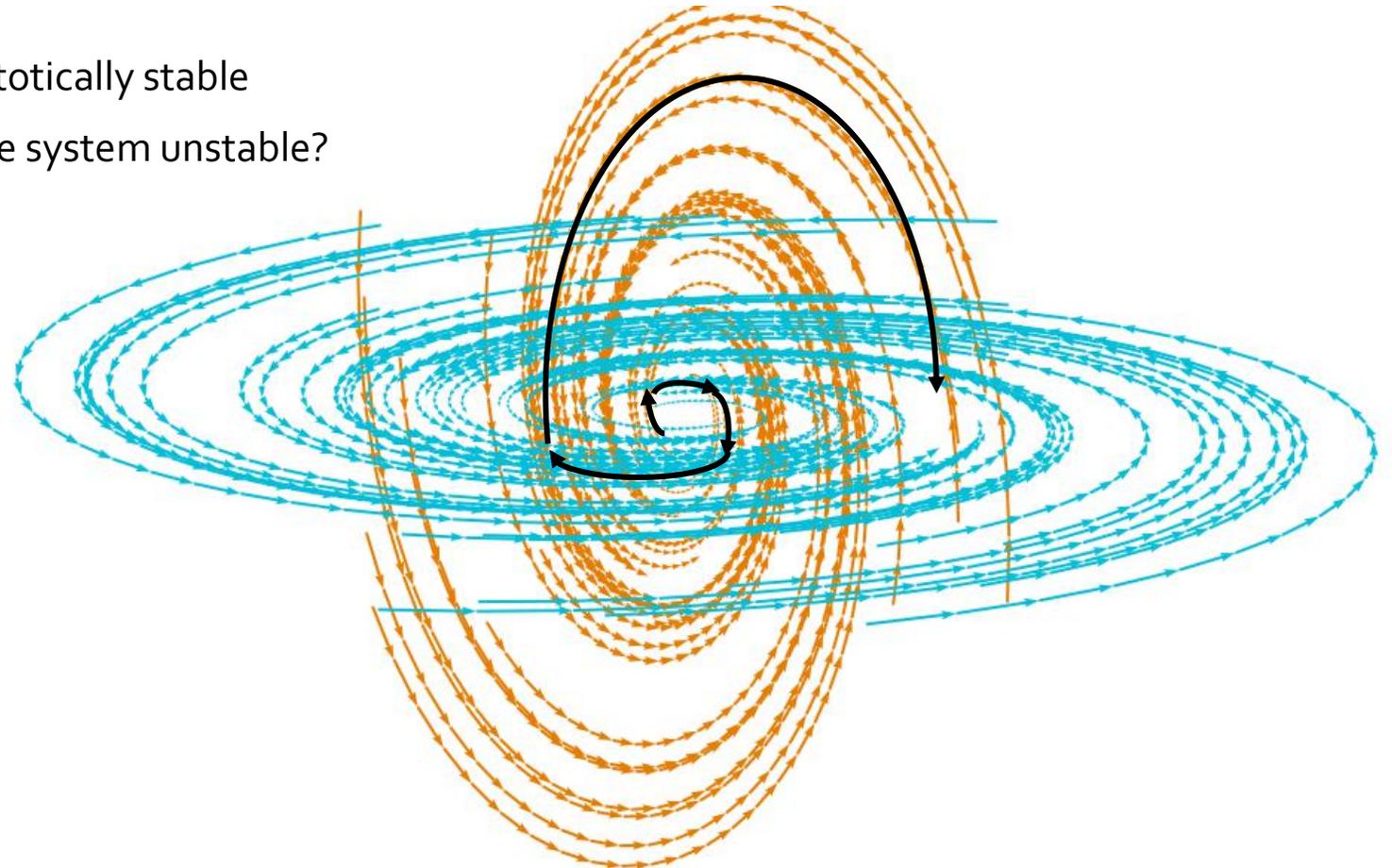
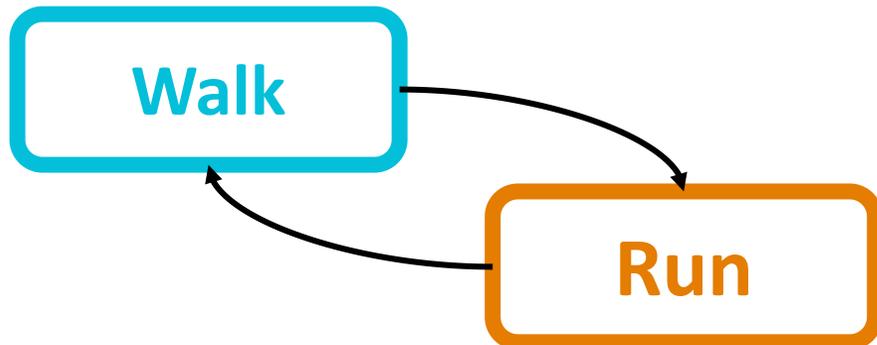
- An hybrid automata is globally uniformly asymptotically stable if:
- For any $\varepsilon > 0$ and any state v_0 , there is a time T such that for any execution fragment α starting from v_0 , for all $t \geq T$, $\|\alpha(t)\| < \varepsilon$

Hybrid Instability

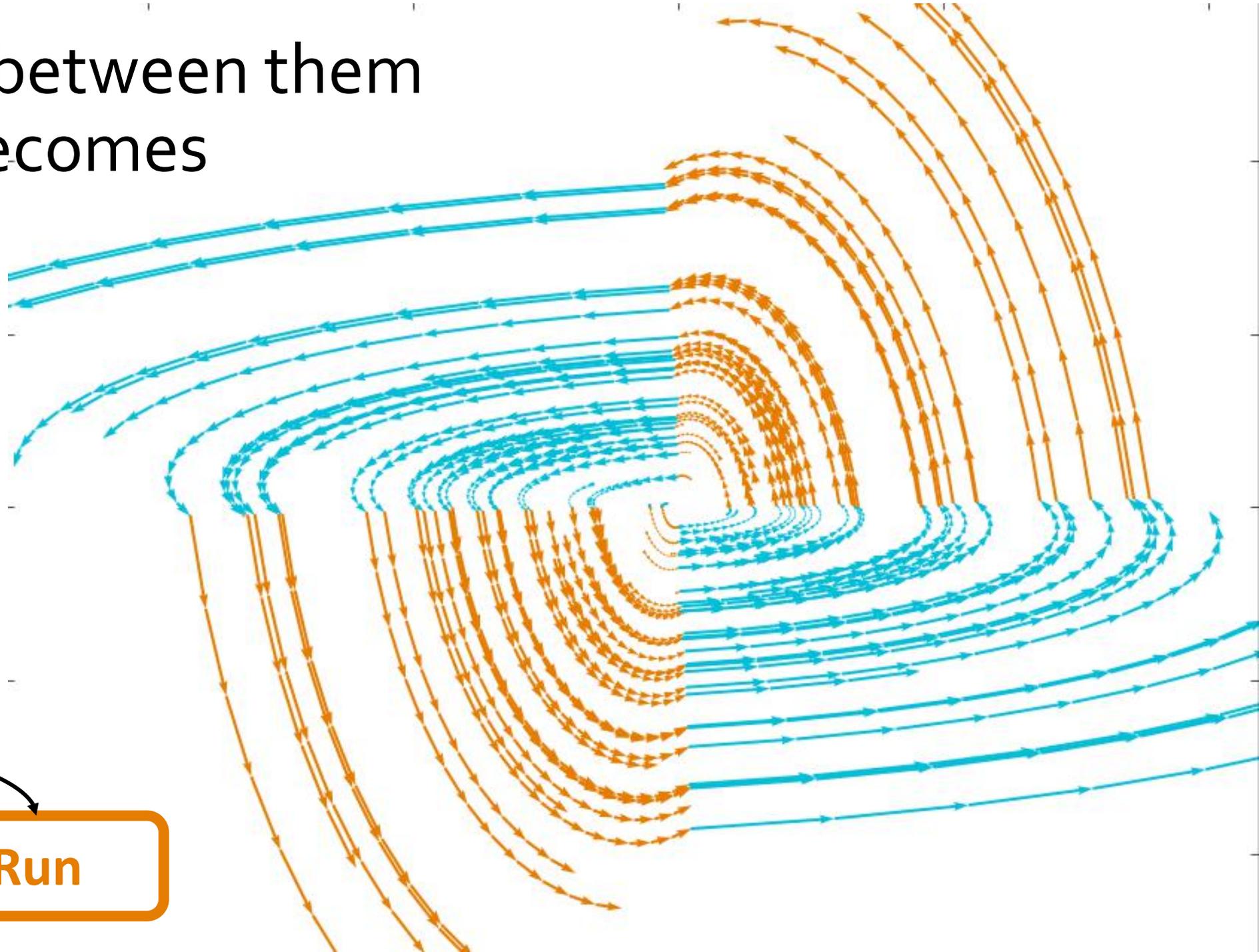
Can we verify the stability of a hybrid system by just verifying the stability of each mode?

Each of the modes of a walking robot are asymptotically stable

Is it possible to switch between them to make the system unstable?



By switching between them
the system becomes
unstable



Walk

Run

Invariants and reachability

- A state x of automaton \mathcal{A} is **reachable** if there exists an execution α with $\alpha.lstate = x$
- $Reach_{\mathcal{A}}(\Theta)$ is the set of all reachable state from Θ
- $Reach_{\mathcal{A}}(\Theta, T)$ is the set of states reachable within time T
- $Reach_{\mathcal{A}}(\Theta, k)$ is the set of states reachable within k transitions
- $Reach_{\mathcal{A}}(\Theta, T, k)$ is the set of states reachable up to time k transitions and time T
- An invariant $I \subseteq val(X)$ is a set of states that contains $Reach_{\mathcal{A}}(\Theta)$