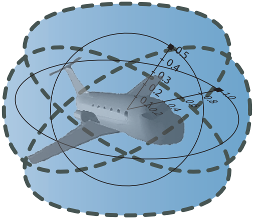
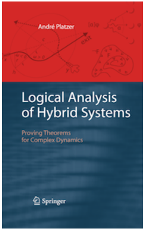
15-819/18-879: Logical Analysis of Hybrid Systems

04: Hybrid Systems Examples

André Platzer

aplatzer@cs.cmu.edu
Carnegie Mellon University, Pittsburgh, PA





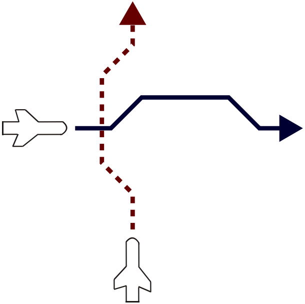
Outline

1. Hybrid Systems Examples
   - Linear Air Traffic Control
   - Bouncing Ball
   - Train Control
   - Water Tank

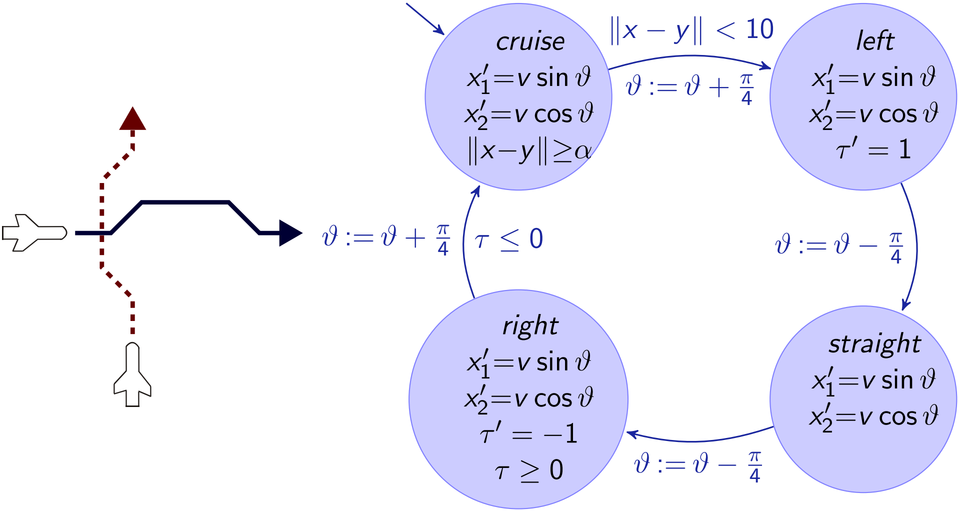
Outline

1. Hybrid Systems Examples
   - Linear Air Traffic Control
   - Bouncing Ball
   - Train Control
   - Water Tank

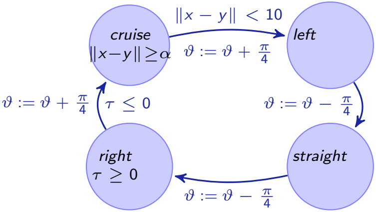
Hybrid Automaton for Collision Avoidance



Hybrid Automaton for Collision Avoidance

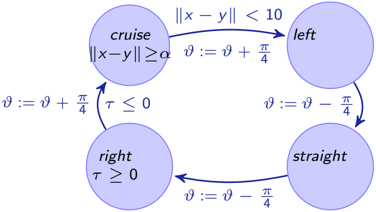


$$\varphi_q(t; x, y, \tau) = \begin{pmatrix} x_1 + tv \sin\vartheta \\ x_2 + tv \cos\vartheta \\ y_1 + tu \sin\varsigma \\ y_2 + tu \cos\varsigma \\ \tau + t \end{pmatrix}$$



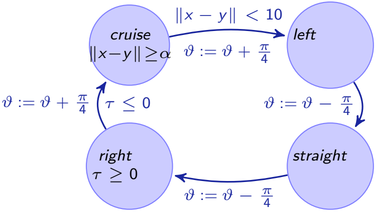
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$



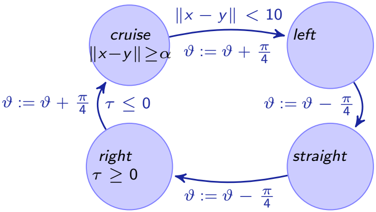
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$



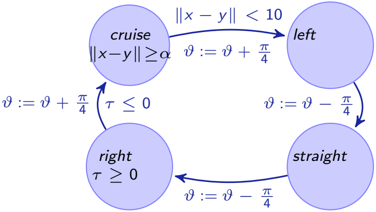
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$



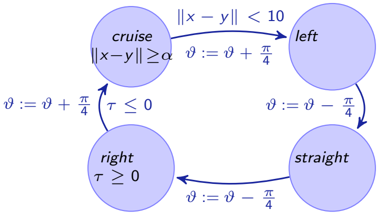
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$
- $jump_e(x, x^+) \equiv guard_e(x) \wedge reset_e(x, x^+)$



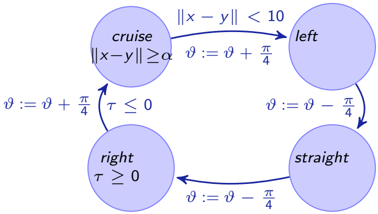
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$
- $jump_e(x, x^+) \equiv guard_e(x) \land reset_e(x, x^+)$
- $guard_{cruise,left} \equiv \|x - y\| < 10$



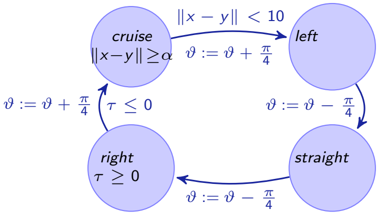
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$
- $jump_e(x, x^+) \equiv guard_e(x) \wedge reset_e(x, x^+)$
- $guard_{cruise,left} \equiv \|x - y\| < 10$
- $reset_{cruise,left} \equiv \vartheta^+ = \vartheta + \frac{\pi}{4}$



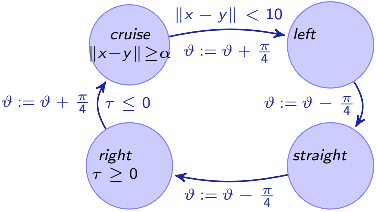
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$
- $jump_e(x, x^+) \equiv guard_e(x) \land reset_e(x, x^+)$
- $guard_{cruise,left} \equiv \|x - y\| < 10$
- $reset_{cruise,left} \equiv \vartheta^+ = \vartheta + \frac{\pi}{4}$
- $reset_{left,straight} \equiv \vartheta^+ = \vartheta - \frac{\pi}{4}$



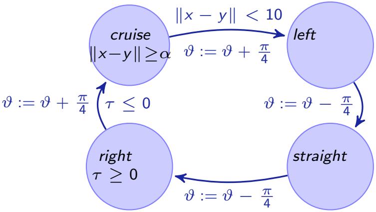
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$
- $jump_e(x, x^+) \equiv guard_e(x) \land reset_e(x, x^+)$
- $guard_{cruise,left} \equiv \|x - y\| < 10$
- $reset_{cruise,left} \equiv \vartheta^+ = \vartheta + \frac{\pi}{4}$
- $reset_{left,straight} \equiv \vartheta^+ = \vartheta - \frac{\pi}{4}$
- $reset_{straight,right} \equiv \vartheta^+ = \vartheta - \frac{\pi}{4}$



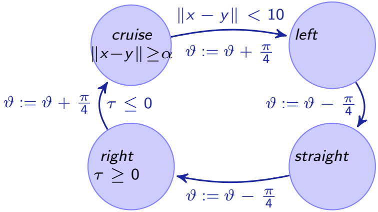
Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$
- $jump_e(x, x^+) \equiv guard_e(x) \wedge reset_e(x, x^+)$
- $guard_{cruise,left} \equiv \|x - y\| < 10$
- $reset_{cruise,left} \equiv \vartheta^+ = \vartheta + \frac{\pi}{4}$
- $reset_{left,straight} \equiv \vartheta^+ = \vartheta - \frac{\pi}{4}$
- $reset_{straight,right} \equiv \vartheta^+ = \vartheta - \frac{\pi}{4}$
- $guard_{right,cruise} \equiv \tau \leq 0$

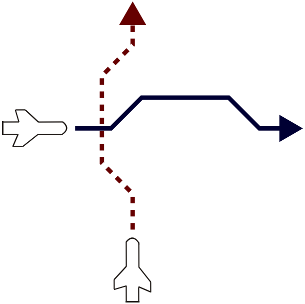


Hybrid Automaton for Collision Avoidance: Formal

- $inv_{cruise} \equiv \|x - y\| \geq \alpha$
- $inv_{right} \equiv \tau \geq 0$
- $inv_{left} \equiv inv_{straight} \equiv true$
- $jump_e(x, x^+) \equiv guard_e(x) \wedge reset_e(x, x^+)$
- $guard_{cruise,left} \equiv \|x - y\| < 10$
- $reset_{cruise,left} \equiv \vartheta^+ = \vartheta + \frac{\pi}{4}$
- $reset_{left,straight} \equiv \vartheta^+ = \vartheta - \frac{\pi}{4}$
- $reset_{straight,right} \equiv \vartheta^+ = \vartheta - \frac{\pi}{4}$
- $guard_{right,cruise} \equiv \tau \leq 0$
- $reset_{right,cruise} \equiv \vartheta^+ = \vartheta + \frac{\pi}{4}$



Hybrid Automaton for Collision Avoidance



Example (Property)

If the aircraft are far apart and have compatible speed, then—when following the protocol—they will never crash?

Example (Property)

If the aircraft enter collision avoidance, then—when following the protocol—will they ever leave again, i.e. follow their old route?

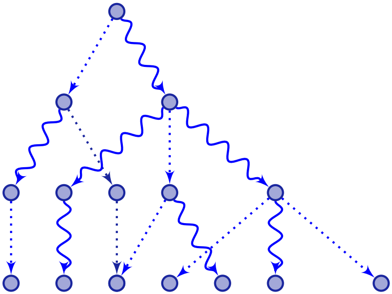
Hybrid Systems

Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \ : \ x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r;x)$ for $r \geq 0$ and $\varphi_q(t;x) \in inv_q$ for all $0 \leq t \leq r$.

Definition (Hybrid Automata → Hybrid System as LTS)

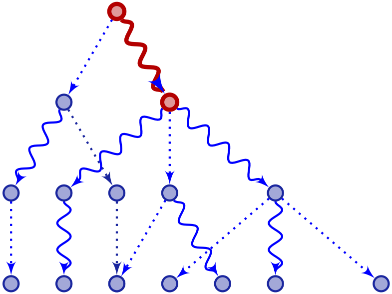
- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q, x) \ : \ x \in inv_q\}$
- Discrete transition $(q, x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q, x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r; x)$ for $r \geq 0$ and $\varphi_q(t; x) \in inv_q$ for all $0 \leq t \leq r$.



Hybrid Systems

Definition (Hybrid Automata → Hybrid System as LTS)

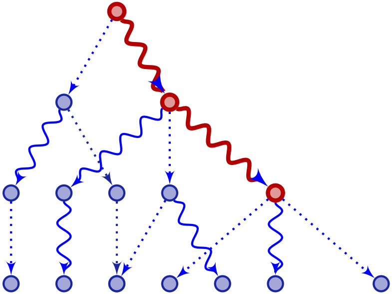
- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \; : \; x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+,x^+)$ iff there is an edge e from q to q^+ with input a such that $(x,x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q,x^+)$ iff $x^+ = \varphi_q(r;x)$ for $r \geq 0$ and $\varphi_q(t;x) \in inv_q$ for all $0 \leq t \leq r$.



Hybrid Systems

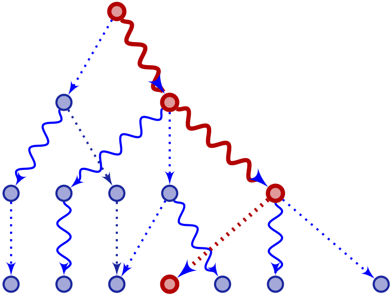
Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q, x) \; : \; x \in inv_q\}$
- Discrete transition $(q, x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q, x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r; x)$ for $r \geq 0$ and $\varphi_q(t; x) \in inv_q$ for all $0 \leq t \leq r$.



Definition (Hybrid Automata $\to$ Hybrid System as LTS)

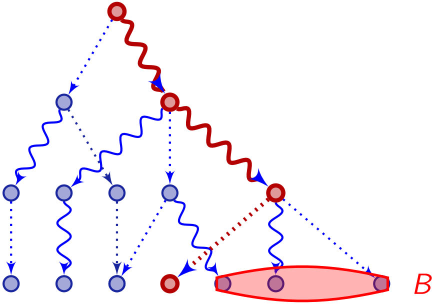
- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \ : \ x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r; x)$ for $r \geq 0$ and $\varphi_q(t; x) \in inv_q$ for all $0 \leq t \leq r$.



Hybrid Systems

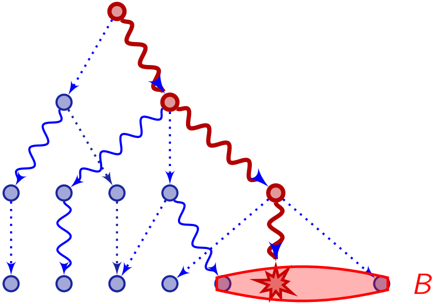
Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \;:\; x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r; x)$ for $r \geq 0$ and $\varphi_q(t; x) \in inv_q$ for all $0 \leq t \leq r$.



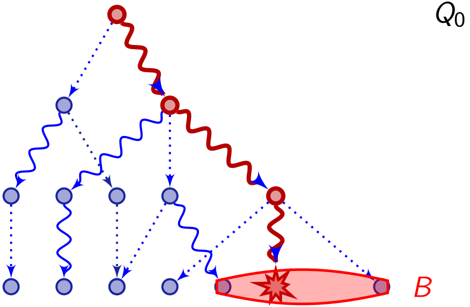
Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \;:\; x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+,x^+)$ iff there is an edge e from q to q^+ with input a such that $(x,x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q,x^+)$ iff $x^+ = \varphi_q(r;x)$ for $r \geq 0$ and $\varphi_q(t;x) \in inv_q$ for all $0 \leq t \leq r$.



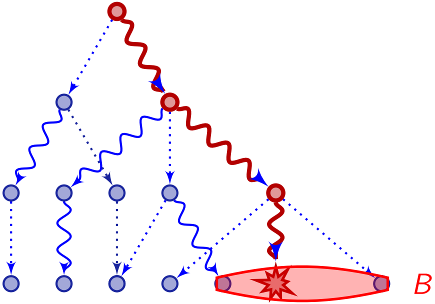
Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \ : \ x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r;x)$ for $r \geq 0$ and $\varphi_q(t;x) \in inv_q$ for all $0 \leq t \leq r$.



Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q, x) \ : \ x \in inv_q\}$
- Discrete transition $(q, x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q, x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r; x)$ for $r \geq 0$ and $\varphi_q(t; x) \in inv_q$ for all $0 \leq t \leq r$.



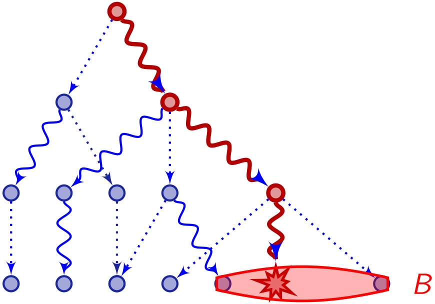
Q_0

$Post_A(Q_0)$

$Q_1 = Post_A(Q_0)$

Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \ : \ x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+,x^+)$ iff there is an edge e from q to q^+ with input a such that $(x,x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q,x^+)$ iff $x^+ = \varphi_q(r;x)$ for $r \geq 0$ and $\varphi_q(t;x) \in inv_q$ for all $0 \leq t \leq r$.



Q_0

$Post_A(Q_0)$

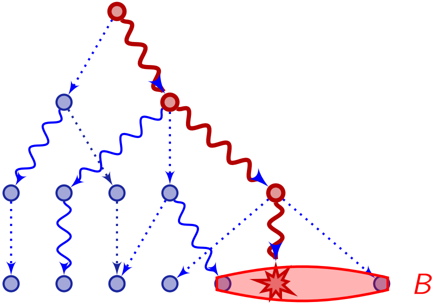
$Q_1 = Post_A(Q_0)$

$Post_A(Q_1)$

$Q_2 = Post_A^2(Q_0)$

Definition (Hybrid Automata → Hybrid System as LTS)

- LTS State space $(Q \times \mathbb{R}^n) \cap \{(q,x) \ : \ x \in inv_q\}$
- Discrete transition $(q,x) \xrightarrow{a} (q^+, x^+)$ iff there is an edge e from q to q^+ with input a such that $(x, x^+) \in jump_e$
- Continuous transition $(q,x) \xrightarrow{r} (q, x^+)$ iff $x^+ = \varphi_q(r;x)$ for $r \geq 0$ and $\varphi_q(t;x) \in inv_q$ for all $0 \leq t \leq r$.



Q_0

$Post_A(Q_0)$

$Q_1 = Post_A(Q_0)$

$Post_A(Q_1)$

$Q_2 = Post_A^2(Q_0)$

$Post_A(Q_2)$

$Q_3 = Post_A^3(Q_0)$

Hybrid Systems Model Checking / Reachability

Definition (Hybrid Systems Model Checking Problem for Reachability)

Given initial states $Q_0 \subseteq Q$ and bad states $B \subseteq Q$ for a hybrid automaton, check whether there is a trace from some $q_0 \in Q_0$ to some $q_b \in B$.

Definition (Hybrid Systems Model Checking Problem for Reachability)

Given initial states $Q_0 \subseteq Q$ and bad states $B \subseteq Q$ for a hybrid automaton, check whether there is a trace from some $q_0 \in Q_0$ to some $q_b \in B$.

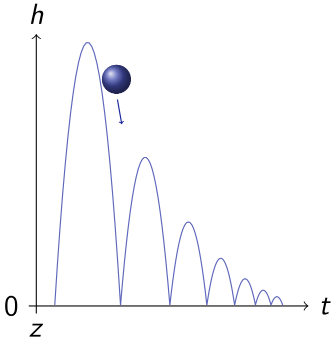
Central question: How to compute?

Definition (Hybrid Systems Model Checking Problem for Reachability)

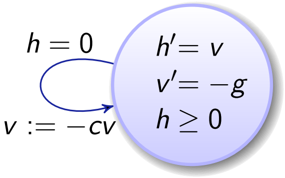
Given initial states $Q_0 \subseteq Q$ and bad states $B \subseteq Q$ for a hybrid automaton, check whether there is a trace from some $q_0 \in Q_0$ to some $q_b \in B$.

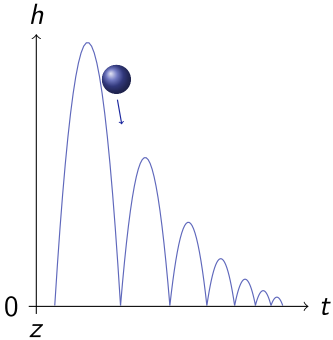
Central question: How to compute? Stay tuned ...

Bouncing Ball as a Hybrid System

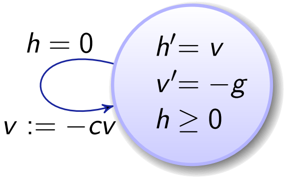


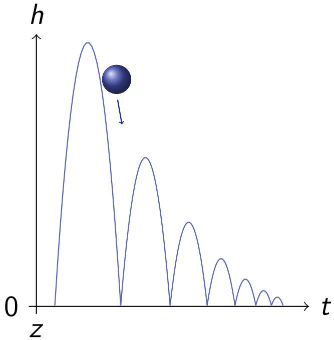
Bouncing Ball as a Hybrid System





Bouncing Ball as a Hybrid System

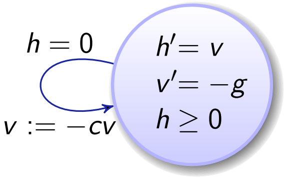


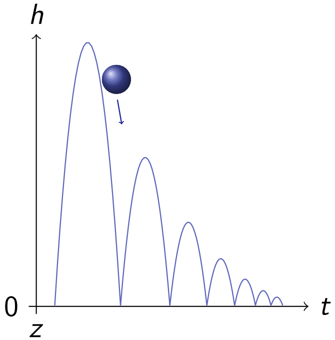


Example (Property)

If initially $h = H$, then bouncing ball always $0 \leq h \leq H$?

Bouncing Ball as a Hybrid System

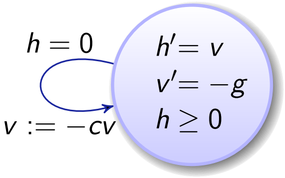


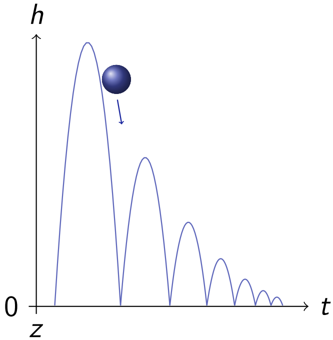


Example (Property)

If initially $h = H$, then bouncing ball always $0 \leq h \leq H$?

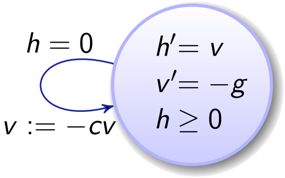
Bouncing Ball as a Hybrid System

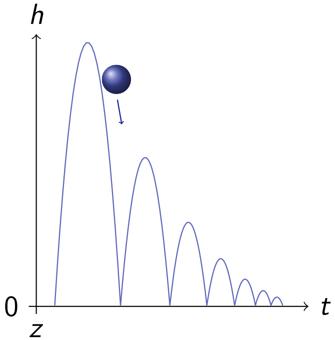




Example (Property)

If initially $h = H$, then bouncing ball always $0 \leq h \leq H$? No!





Example (Property)

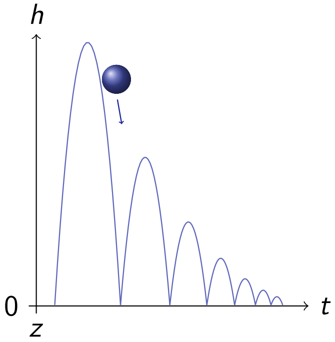
If initially $h = H$, then bouncing ball always $0 \leq h \leq H$? **No!**

Initial $v > 0$ then climbs first. Does bouncing ever stop?

Aerodynamical Bouncing Ball as a Hybrid System

There is wind resistance, so the ODE should be

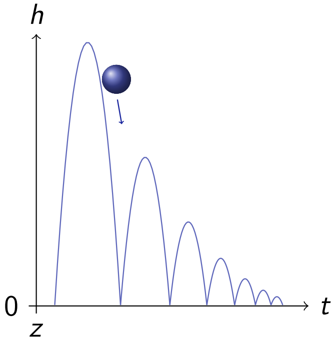
$$h' = v, v' = -g$$



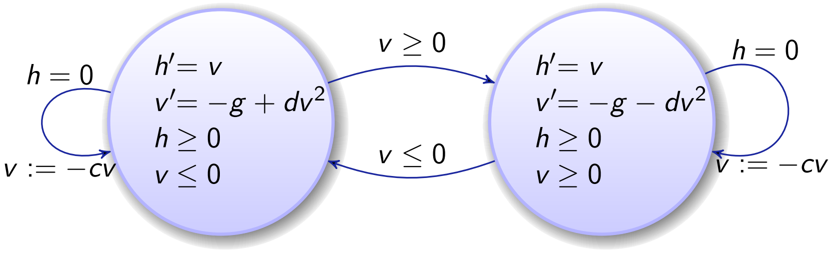
Aerodynamical Bouncing Ball as a Hybrid System

There is wind resistance, so the ODE should be

$$h' = v, v' = -g$$



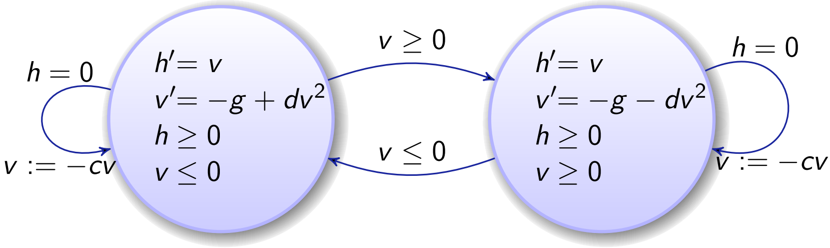
Aerodynamical Bouncing Ball as a Hybrid System



Example (Property)

If initially $h = H$, then bouncing ball always $0 \leq h \leq H$?

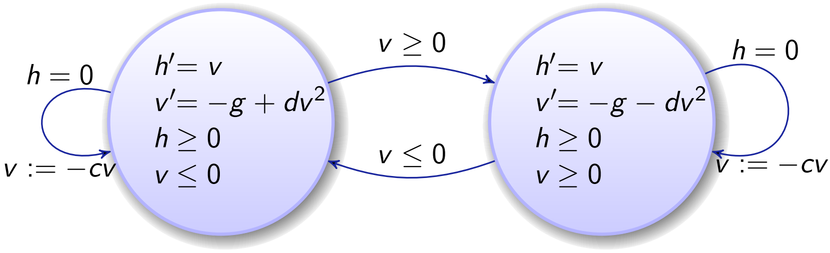
Aerodynamical Bouncing Ball as a Hybrid System



Example (Property)

If initially $h = H$, then bouncing ball always $0 \leq h \leq H$?

Aerodynamical Bouncing Ball as a Hybrid System



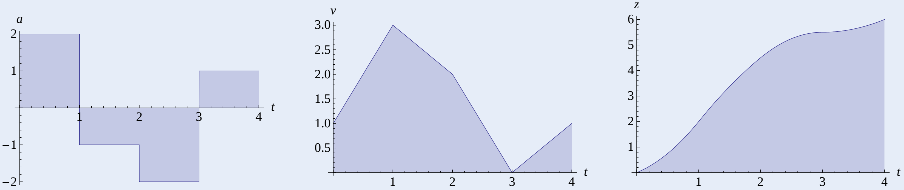
Example (Property)

If initially $h = H$, then bouncing ball always $0 \leq h \leq H$? No!

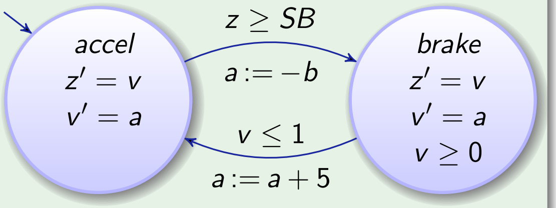
Challenge

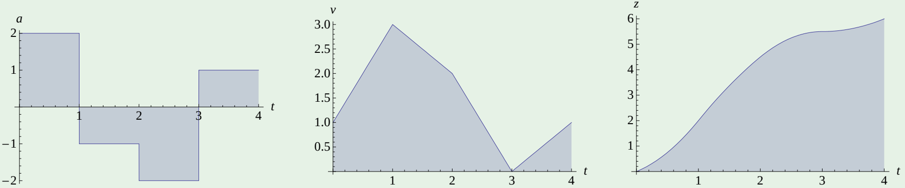
Hybrid systems

- Continuous dynamics (differential equations)
- Discrete dynamics (control decisions)

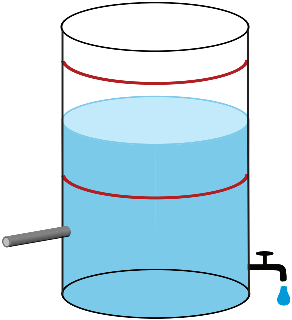


Example (Overly Simplistic Train Control)

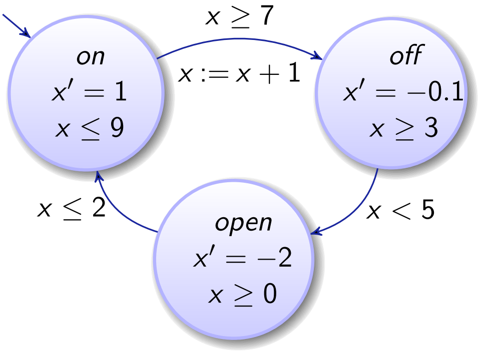


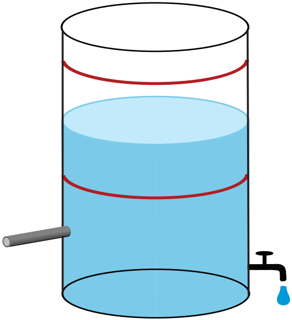


Simple Water Tank

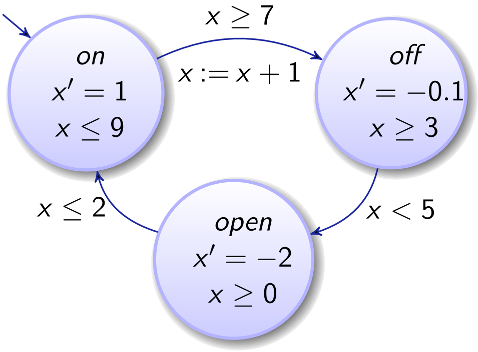


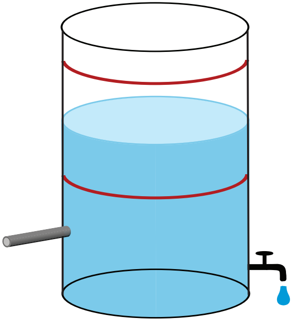
Simple Water Tank





Simple Water Tank

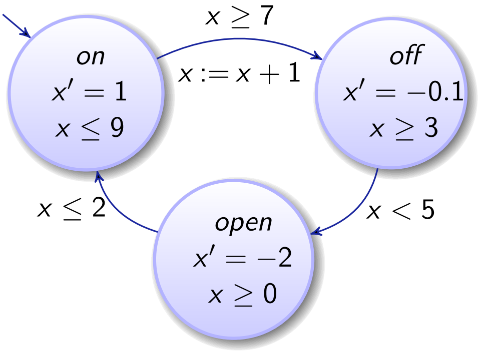


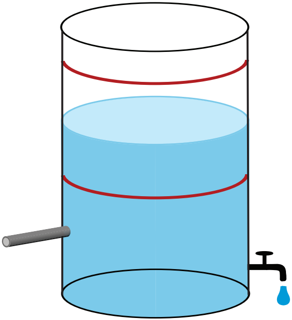


Example (Property)

If initially $1 \leq x \leq 10$, then water tank always $1 \leq x \leq 10$?

Simple Water Tank

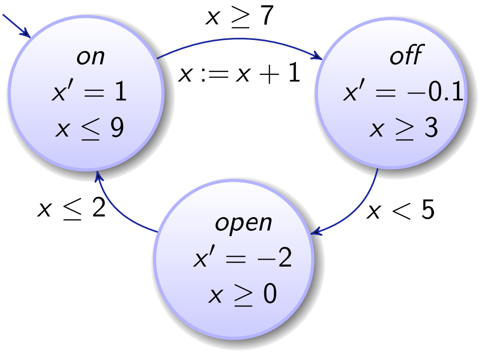


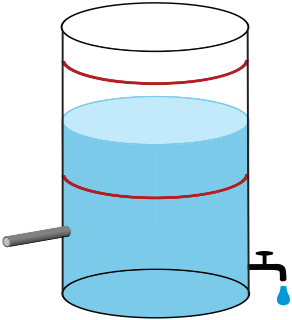


Example (Property)

If initially $1 \leq x \leq 10$, then water tank always $1 \leq x \leq 10$?

Simple Water Tank

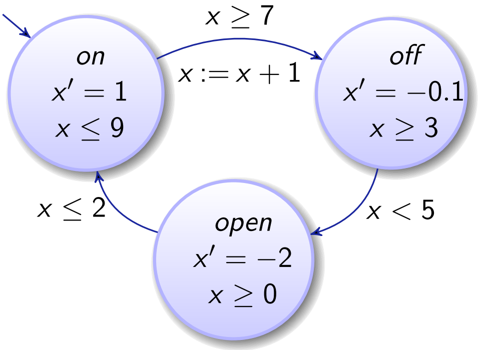


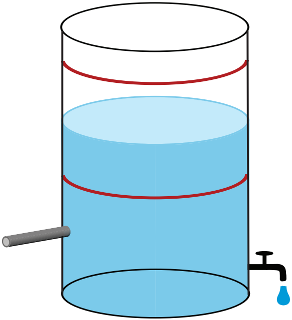


Example (Property)

If initially $1 \leq x \leq 10$, then water tank always $1 \leq x \leq 10$? No!

Simple Water Tank





Example (Property)

If initially $1 \leq x \leq 10$, then water tank always $1 \leq x \leq 10$? No!

Can stay in *open* too long, even until $x = 0$

T. Krilavičius.
Bestiarium of hybrid systems.

A. Platzer.
Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics.
Springer, Heidelberg, 2010.