Hybrid System

Discrete system

Discrete system:

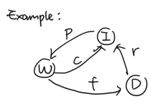
$q \in Q = \{I, W, D\}$. Q: state space ⓘ } finite sets

$\Sigma = \{p, c, f, r\}$. Σ: event space ⓘ

The state after an event occurs is given by a transition relation ⓘ

$\delta : Q \times \Sigma \to Q$

Example:



$\delta(I, p) = W$

$\delta(W, c) = I$

$\delta(W, f) = D$

$\delta(D, r) = I$

Transition system & FSM

Transition system $T = (S, \delta, S_0, S_F)$

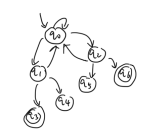
S: set of states (can be finite or infinite) hybrid automaton

δ: transition relation $\delta: S \to P(S)$ {set of subsets of S} $\equiv 2^S$

$S_0 \subset S$: set of initial states

$S_F \subset S$: set of final states

FSM is a transition system with S finite



$T = (S, \delta, S_0, S_F)$

$S = \{q_0, \cdots q_6\}$

$S_0 = \{q_0\}$

$S_F = \{q_3, q_6\}$

$\delta(q_0) = \{q_0, q_1, q_2\}$

$\delta(q_1) = \{q_0, q_3, q_4\}$

Model checking

Model checking.

Automatically analyze & check properties of a system by exploring the state space.

Finite state systems

- Properties can be verified by exploring all states.
- Since the system has a finite number of states, it is guaranteed that the process ends.

Hybrid systems.

- Infinite number of states
- "Equivalent" to finite state systems.

Nonlinear systems. $\dot{x} = f(x)$, x_0, $x \in X$

$T = (S, \delta, S_0, S_F)$

$S = X$

$S_0 = \{x_0\} = X_0$

$S_F = S$

$\delta(x) = \{x' \in X \mid \exists T > 0, x(T) = x'\}$

↑ solution of $\dot{x} = f(x)$, $x_0 = T$

Safety & Liveness

Safety property.

the state (q, x) always remains in a set of states $F \subset Q \times X$ ("not bad set")

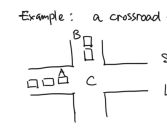
temporal logic $\square (q, x) \in F$

Liveness property.

the state (q, x) certainly reaches (eventually) a set of states $\bar{F} \subset Q \times X$ ("good set")

temporal logic $\lozenge (q, x) \in \bar{F}$

Example: a crossroad traffic controller



Safety: never A crosses ∧ B crosses

Liveness: eventually all cars get through C

Theorem

Given a hybrid system H (/ an automaton / finite-state system FSM) meets a safety specification $\square(q, x) \in F$

iff $Reach_H \subset F$

↑ a set of states reachable from the initial states.

↓ if $Reach_H \subset F \Rightarrow Init \subset F$

Sequence properties (of hybrid automata)

Sequence property: $P: Q_{sig} \times X_{sig}$

X_{sig} = set of all piecewise continuous signals $x: [0, T) \to \mathbb{R}^n$, $T \in (0, \infty)$

Q_{sig} = set of all piecewise constant signals $q: [0, T) \to Q$, $T \in (0, \infty)$

$P: Q_{sig} \times X_{sig} \to \{false, true\}$

$P(q, x) = \begin{cases} true & q(t) \in \{\cdot\} \wedge x(t) \in \{\cdot\}, \forall t \\ false & otherwise \end{cases}$ a condition of (q, x)

a pair of signals (q, x) satisfies P if $P(q, x) = true$.

a hybrid automaton H satisfies P: $H \models P$, if $P(q, x) = true$ for every execution of H.

Pb: Given a H, and a sequence property P, show if $H \models P$.

$(\square P)(t_0) \iff \forall t \geq t_0$, $P(t) = P(q(t), x(t))$ holds true. (always)

$(\lozenge P)(t_0) \iff \exists t \geq t_0$, $P(t)$ is true (eventually)

$(\bigcirc P)(t_0) \iff P(t_0^+)$ is true (next time)

$(\square \lozenge P)(t_0) \iff \forall t_1 \geq t_0, \exists t \geq t_1, P(t)$ true

$(\lozenge \square P)(t_0) \iff \exists t_1 \geq t_0, \forall t \geq t_1, P(t)$ true

Example: bouncing ball:

$\square \{x_1 \geq 0\}$

$\lozenge \{x_1 = 0\}$

$\square \lozenge \{x_1 = 0\}$

Safety & Liveness property

$x^*: [0, T^*) \to \mathbb{R}^n$ is a PREFIX of x if $T^* \leq T$ and $x^*(t) = x(t)$, $\forall t \in [0, T^*)$

Safety property is a sequence property:

1. nonempty $\exists (q, x): P(q, x)$ true (nontrivial)
2. prefix closed given (q, x), $P(q, x) \Rightarrow P(q^*, x^*)$ a prefix to a "good" signal is also good
3. limit closed $(q_1, x_1), (q_2, x_2), \ldots, (q_k, x_k)$ prefix of (q_{k+1}, x_{k+1}) $\forall k$, satisfies P, then $\lim_{k \to \infty} (q_k, x_k) = (q, x)$ also satisfies P. If something bad happens, it will happen in finite time.

Liveness property is a sequence property such that for ∀ finite $(q^*, x^*) \in Q_{sig} \times X_{sig}$ there are some (q, x):

1. (q^*, x^*) is a prefix to (q, x), and
2. (q, x) satisfies P

Something "good" will eventually happen.

Theorem: if P is both a liveness and a safety property then P is always true.

Theorem: for every nonempty sequence property P, there exist a safety property P_1 and a liveness property P_2: (q, x) satisfies $P \iff (q, x)$ satisfies both P_1 & P_2

Reachability

Trajectory

A trajectory of a transition system is a finite or infinite sequence of states s_i s.t. ① $s_0 \in S_0$ ② $s_{i+1} \in \delta(s_i)$ for $\forall i = 0, \cdots, N-1$

Reachability.

Given a transition system T, the set of final states S_F is called reachable if ∃ a trajectory of T, s.t. $s_i \in S_F$, for some i.

Check reachability

Operator Pre(·) [Predecessor]

$Pre: P(S) \to P(S)$

$Pre(\hat{S}) = \{s \in S \mid \exists \hat{s} \in \hat{S} \mid \hat{s} \in \delta(s)\}$

Algorithm: backwards reachability

$W_0 = S_F$, $i = 0$ (initialization)

repeat

if $(W_i \cap S_0 \neq \emptyset)$ return "S_F reachable"

endif

$W_{i+1} = Pre(W_i) \cup W_i$

$i = i + 1$

until $W_i = W_{i-1}$

return "S_F not reachable" → in final W_i, $Pre(W_i) = W_i$, $W_i \cap S_0 = \emptyset$

this algorithm does not necessarily end.

Bisimulations

Bisimulation is the useful Partition to check Reachability

Bisimulations.

Idea: Group states that have "similar" behavior.

Definition: a partition of S is a sequence of sets of states $\{S_i\}_{i \in I}$ such that: $\bigcup_{i \in I} S_i = S$, $S_i \cap S_j = \emptyset$, $\forall i \neq j$

the partition is finite if I is finite, it is infinite if I is infinite.

Definition: a bisimulation of a transition system $T = (S, \delta, S_0, S_F)$ is a partition of the state space S s.t.:

① S_0 is the union of elements of $\{S_i\}_{i \in I}$

② S_F is the union of elements of $\{S_i\}_{i \in I}$

③ $\forall i, j \in I$, $\forall s, \hat{s} \in S_i$, if $\delta(s) \cap S_j \neq \emptyset$ then $\delta(\hat{s}) \cap S_j \neq \emptyset$

If there is any s in S_i has a transition to S_j, then any other $\hat{s}$ in S_i has a transition there too.

Quotient transition system

Given a transition system $T = (S, \delta, S_0, S_F)$ and a partition of the state space $\{S_i\}_{i \in I}$, we can define a transition system whose states are the elements of the partition $S_i \subseteq S$ (rather than individual states $s \in S$). This system $\hat{T} = (\hat{S}, \hat{\delta}, \hat{S}_0, \hat{S}_F)$ is defined as:

① $\hat{S} = \{S_i\}_{i \in I}$, i.e. the state space is the partition.

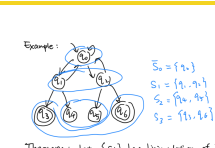
② $\hat{\delta}$ allows a transition from S_i to S_j iff δ allows a transition from some states in S_i to some other states in S_j: $\hat{\delta}(S_i) = \{S_j \mid \exists s \in S_i, \exists s' \in S_j, s.t.\ s' \in \delta(s)\}$

③ S_i is in the initial set of $\hat{T}$, i.e. $S_i \in \hat{S}_0$ iff some elements in S_i ($s \in S_i$) is an initial state of the original transition system ($s \in S_0$)

④ S_i is in the final set of $\hat{T}$, i.e. $S_i \in \hat{S}_F$ iff $\exists s \in S_i$, s.t. $s \in S_F$

Quotient transition system wrt bisimulation

Example:



$S_0 = \{q_0\}$, $S_1 = \{q_1, q_2\}$, $S_2 = \{q_5, q_6\}$, $S_3 = \{q_3, q_4\}$

$\hat{T} = (\hat{S}, \hat{\delta}, \hat{S}_0, \hat{S}_F)$

$\hat{S} = \{S_0, S_1, S_2, S_3\}$

$\hat{\delta}(S_0) = \{S_0, S_1, S_2\}$

$\hat{S}_0 = \{S_0\}$

$\hat{S}_F = \{S_3\}$

Theorem: let $\{S_i\}_{i \in I}$ be a bisimulation of transition system T and let $\hat{T}$ be the corresponding quotient transition system w.r.t. the bisimulation then: S_F is reachable $\iff \hat{S}_F$ is reachable (in $\hat{T}$)

stronger results: bisimulations preserve all other properties that can be expressed in temporal logic.

Bisimulation algorithm

Automaton is a transition with infinite states

Hybrid automaton

Hybrid automata = (Autonomous) hybrid systems.

$H = (Q, X, Init, f, Dom, E, G, R)$

Q: finite set of discrete states

$X \subseteq \mathbb{R}^n$: set of continuous states

$Init \subseteq Q \times X$: set of admissible initial states

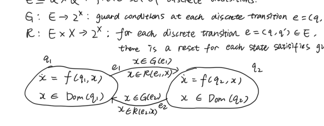
$f: Q \times X \to TX$: the continuous flow in each discrete state $q \in Q$: $\dot{x} = f(q, x)$

$Dom: Q \to 2^X$: for each $q \in Q$, there is a domain for x: $x \in Dom(q)$

$E \subseteq Q \times Q$: finite set of discrete transitions.

$G: E \to 2^X$: guard conditions at each discrete transition $e = (q, q') \in E$

$R: E \times X \to 2^X$: for each discrete transition $e = (q, q') \in E$, there is a reset for each state satisfies guard condition $x \in G(e)$



2^X: set of subsets of X

Rectangular set

Consider $x \in \mathbb{R}^n$. A subset of $\mathbb{R}^n$ set is called rectangular if it can be written as a finite boolean combination of constraints of the form: $x_i \leq a$, $x_i < b$, $x_i = c$, $x_i \geq d$, $x_i > e$, where a, b, c, d, e are rational numbers. Roughly speaking, rectangular sets are "rectangles" in $\mathbb{R}^n$ whose sides are aligned with the axes, or unions of such rectangles.

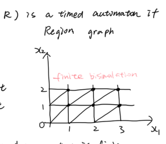
Examples: ① $x_1 \geq 0$ ② $(x_1 \geq 0) \wedge (x_1 \leq 2)$ ③ $\emptyset$

Timed automata

$H(Q, X, f, Init, D, E, G, R)$ is a timed automaton if

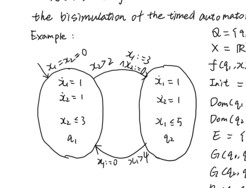
- Q, X, E
- $f(q, x) = 1$ clock
- Init singleton set
- Dom(·) rectangular set
- G(·) rectangular set
- R(·,·) singleton set

Region graph



the bisimulation of the timed automaton is finite.

Example:



$Q = \{q_1, q_2\}$

$X = \mathbb{R}^2$

$f(q_1, x) = f(q_2, x) = [1\ 1]$

$Init = \{(q_1, (0, 0))\}$

$Dom(q_1) = \{x \in \mathbb{R}^2 \mid x_2 \leq 3\}$

$Dom(q_2) = \{x \in \mathbb{R}^2 \mid x_1 \leq 5\}$

$E = \{(q_1, q_2), (q_2, q_1)\}$

$G(q_1, q_2) = \{x \in \mathbb{R}^2 \mid x_2 \geq 2\}$

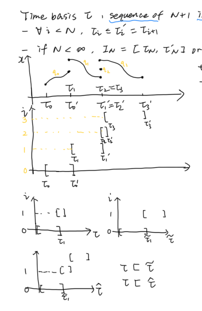
$G(q_2, q_1) = \{x \in \mathbb{R}^2 \mid x_1 \geq 4\}$

$R(q_1, q_2, x) = \{(3, 0)\}$ $R(q_2, q_1, x) = \{(0, x_2)\}$

Time basis

Time basis τ: sequence of N+1 intervals $I_i = [\tau_i, \tau_i']$ ($i = 0, \ldots, N$) s.t.

- $\forall i < N$, $\tau_i \leq \tau_i' = \tau_{i+1}$
- if $N < \infty$, $I_N = [\tau_N, \tau_N']$ or $[\tau_N, \tau_N')$



$t_1 \in [\tau_i, \tau_i'] \in \tau$, $t_2 \in [\tau_j, \tau_j'] \in \tau$

t_1 precedes t_2: $t_1 \prec t_2$ if $t_1 < t_2$ or $i < j$

set of all time basis $\mathcal{T}$: $\tau \in \mathcal{T}$, $\hat{\tau} \in \mathcal{T}$

PREFIX:

$\tau = \{I_i\}_{i=0}^N$ is said to be a prefix of $\hat{\tau} = \{\hat{I}_i\}_{i=0}^M$, i.e. $\tau \sqsubseteq \hat{\tau}$, if either $\tau = \hat{\tau}$ or $N \leq M$, $I_i = \hat{I}_i$, $i = 0, \ldots, N-1$, $I_N \subseteq \hat{I}_N$

a strict prefix of $\hat{\tau}$, i.e. $\tau \sqsubset \hat{\tau}$, if τ is a prefix but $\tau \neq \hat{\tau}$

Finite, infinite or zeno time basis

Length of time basis.

the "discrete" length = N

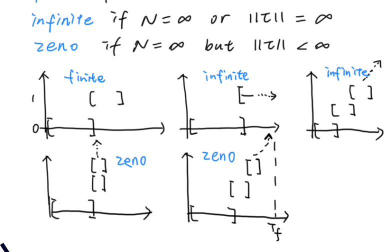
the "continuous" length $= \|\tau\| = \sum_{i=0}^{N} (\tau_i' - \tau_i)$

a time basis is

finite if $N < \infty$ and $I_N = [\tau_N, \tau_N']$

infinite if $N = \infty$ or $\|\tau\| = \infty$

zeno if $N = \infty$ but $\|\tau\| < \infty$



Execution

Execution (or evolution) of a hybrid automaton $H = (Q, X, Init, f, Dom, E, G, R)$ is a trajectory $\chi = (\tau, q, x)$, $q = \{q_i\}_{i=0}^N$, $x = \{x^i\}_{i=0}^N$ with $\tau \in \mathcal{T}$, $q_i: I_i \to Q$, $x^i: I_i \to x^i(t)$ $I_i = [\tau_i, \tau_i']$

such that

- $(q_0(0), x^0(0)) \in Init$ define a initial set
- Discrete evolution
  - $\forall i$ $(q_i(\tau_i'), q_{i+1}(\tau_{i+1})) \in E$ q changes
  - $x^i(\tau_i') \in G(q_i(\tau_i'), q_{i+1}(\tau_{i+1}))$ still need to check guard condition
  - and $x^{i+1}(\tau_{i+1}) \in R(q_i(\tau_i'), q_{i+1}(\tau_{i+1}), x^i(\tau_i'))$ x resets
- Continuous evolution
  - $q_i(\cdot): I_i \to Q$ is constant over $t \in I_i$: $q_i(t) = q_i(\tau_i)$ q constant
  - $x^i(\cdot): I_i \to X$ solution of $\frac{dx^i}{dt} = f(q_i(t), x^i(t))$ over I_i, starting from $x^i(\tau_i)$ $\dot{x} = f(\cdot)$ need an initial value
  - $x^i(t) \in Dom(q_i(t))$ $\forall t \in [\tau_i, \tau_i')$ still need to check domain condition.

Execution time

An execution time $\tau_\infty(\chi)$ of a trajectory χ is

$\tau_\infty(\chi) = \sum_{i=0}^{N} (\tau_i' - \tau_i)$

a trajectory χ is

Finite if $N < \infty$ and $I_N = [\tau_N, \tau_N']$

Infinite if $N = \infty$ or $\tau_\infty(\chi) = \infty$

Zeno if $N = \infty$ but $\tau_\infty(\chi) < \infty$

Maximal if it is not the strict prefix of any other execution.

Reach & Trans

$\Omega_H(q_0, x_0)$: set of all trajectories of H with $(q_0, x_0) \in Init$

$\Omega_H^\infty(q_0, x_0)$: set of all infinite trajectories of H with $(q_0, x_0) \in Init$

$\Omega_H = \bigcup_{(q_0, x_0) \in Init} \Omega_H(q_0, x_0)$; $\Omega_H^\infty = \bigcup_{(q_0, x_0) \in Init} \Omega_H^\infty(q_0, x_0)$

Reachable states

$Reach_H = \{(\hat{q}, \hat{x}) \in Q \times X : \exists \chi \in \Omega_H, (q(\tau_N'), x(\tau_N')) = (\hat{q}, \hat{x}), N < \infty\}$

Transition states

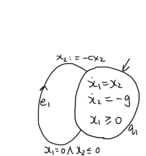
$Trans_H = \{(q, x) \in Q \times X : \forall \varepsilon > 0, \exists t \in [0, \varepsilon): \Phi(t, x) \notin Dom(q)\}$

when the solution x(t) has jumped from Dom(q); Φ: solution of the differential equation associated with q and initial condition x.

$\bigcup_{q \in Q} \{q\} \times Dom(q)^c \subseteq Trans_H$ (complement set)

if Dom(q) is a closed set, Trans may also contain pieces of the boundary. (x(t) is about to jump)

Example



$Reach_H = Init = q_1 \times \{x_1 \geq 0\}$

$Trans_H = \{q_1 \times \{x_1 < 0\}\} \cup \{q_1 \times \{x_1 = 0, x_2 \leq 0\}\}$

$Trans_H \cap Reach_H = \{q_1 \times \{x_1 = 0, x_2 \leq 0\}\}$

Non-blocking & deterministic

Def: H is non-blocking if for ∀ initial states $(q_0, x_0) \in Init$, s.t. $\Omega_H^\infty(q_0, x_0) \neq \emptyset$, i.e. ∃ an infinite execution starting at (q_0, x_0)

Def: H is deterministic if for ∀ initial states $(q_0, x_0) \in Init$, s.t. there ∃ at most one maximal execution χ

Theorem: H is non-blocking if

(#) $\forall (\hat{q}, \hat{x}) \in Reach \cap Trans$, $\exists \hat{q}' \in Q$: $(\hat{q}, \hat{q}') \in E$ and $\hat{x} \in G(\hat{q}, \hat{q}')$

explanation: for ∀ reachable transition states $(\hat{q}, \hat{x})$, ∃ an execution from it to $(\hat{q}', \hat{x}')$ and $\hat{x}$ satisfies the guard condition for $e = (\hat{q}, \hat{q}')$

If H is deterministic, then it is non-blocking iff condition (#) holds.

Proof: blabla

Theorem: H is deterministic iff

- if $(q, q') \in E$ and $x \in G(q, q')$ then $(q, x) \in Trans_H$ ①
- if $(q, q') \in E$ and $(q, q'') \in E$ with $q' \neq q''$ ② then $x \notin G(q, q') \cap G(q, q'')$
- if $(q, q') \in E$ and $x \in G(q, q')$ then $R(q, q', x)$ returns only one element. ③

Hint: nondeterminism is due to ① non-deterministic resets of x ② non-deterministic transitions

Existence and uniqueness of executions

H accepts a unique infinite execution for ∀ initial state if H is non-blocking & deterministic. ☆

Theorem: Existence of solutions

If $f: \mathbb{R}^n \to \mathbb{R}^n$ is continuous

$\Rightarrow \forall x_0 \in \mathbb{R}^n$, ∃ at least one solution with $x(0) = x_0$ in $[0, \varepsilon)$

Theorem: Uniqueness

If $f: \mathbb{R}^n \to \mathbb{R}^n$ is Lipschitz continuous

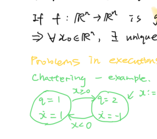
$\Rightarrow \forall x_0 \in \mathbb{R}^n$, ∃ unique solution with $x(0) = x_0$ in $[0, \varepsilon)$

If $f: \mathbb{R}^n \to \mathbb{R}^n$ is globally Lipschitz continuous,

$\Rightarrow \forall x_0 \in \mathbb{R}^n$, ∃ unique solution with $x(0) = x_0$ in $[0, \infty)$

Problems in executions deriving from the continuous evolution.

Chattering — example.



$\tau = \{[0, 1], [1, 1], [1, 1], [1, 1], \ldots\}$

τ is infinite but after [0, 1] all intervals are [1, 1]