Hybrid systems (H.S.)

1. Modeling of Hybrid Systems
2. Examples of ""
3. Variable structure systems
4. Complementarity systems
5. Analysis of H.S.
6. Control Design of H.S.
7. Miscellaneous topics (time permitting)

1. Modeling of H.S.

1.1. Introduction
   "Definition"
   Generally speaking, discrete events interact
   continuous dynamics.

Three views of H.S.

- Computer science: discrete (computer program)
- Interacting with analog environment (continuous dynamics)
- Key issue: verification

Modeling & simulation community:
- Multi-modal system simulation
- Key issue: existence and uniqueness of solutions.
Systems and Control:

a) switching and relay control: power electronics, queuing systems (communication networks, traffic networks, manufacturing systems etc.)

b) supervisory or hierarchical control

c) discontinuous control for some nonlinear systems (e.g. shift free systems, regulation problem)

Definition of Hybrid Systems

Behavior modeling in terms of all possible trajectories of continuous and discrete variables associated with the system.

Hybrid automaton model "working definition" based on construction.

Dynamics

Continuous domain "ode"  \[ \text{in general: pde, stochastic ode, etc.} \]

Discrete domain "finite automata"  \[ \text{pushdown automata, turing machines, etc.} \]

Continuous-time state-space model

State variables \( x \in \mathbb{R}^n \) (or \( n \)-dimensional manifold \( X \))

External variables \( w \in \mathbb{R}^8 \)

Relation: \[ F(x, \dot{x}, w) = 0 \] differential and algebraic equations
Solutions of $0$ are "sufficiently smooth" $x(t)$ and $w(t)$ satisfying

$$F(x(t), x(t), w(t)) = 0$$

for (almost) all times $t \in \mathbb{R}$ (continuous time axis)

The advantage of the general model over ISO is that the "general model" is closed over interconnections.

We can allow $w(t) \in PWC$ (allowing discontinuities with $u(t)$)

and $x(t) \in C$ and $u$ differentiable with

satisfied at all points except at discontinuities of $w(t)$ and non differentiability of $x(t)$.

Finite Automaton: Triple $(L, A, E)$.

$L$: Finite state space

$A$: Finite alphabet set with elements "symbols"

$E$: Transition rule set

$$E \subseteq L \times A \times L$$

All elements are called edges (or transitions or events).

Path (or trajectory): $(l_0, a_1, l_1, a_2, \ldots, l_n, a_{n+1}, l_{n+1})$

with $(l_i, a_i, l_{i+1}) \in E$ for $i = 1, 2, \ldots, n-1$
Graph representation of $F^+$.

\[ a \rightarrow b \rightarrow c \rightarrow l_n \rightarrow b \]
\[ l_1 \rightarrow d \rightarrow l_2 \rightarrow a \]

Vertices given by elements of $L$

Edges "" "" $E$

Labels "" "" A Cabo called

synchronization labels sometimes

since interconnection with other automata

take place via true "shared" symbols.

Input-output automata:
\[ \text{Two symbols associated with each edge} \]

input symbol $i$, and
output symbol $o$

Deterministic automata have only one edge
from a given state with the same input symbol.

Deterministic input-output automata can be represented by
\[ l^# = v(l, i) \]
\[ o = \eta(l, i) \]

Model may include ICL (initial state set) and FCL (final state subset).

: A successful path is
\[ (l_0, a_0, l_1, a_1, \ldots, l_n, a_{n-1}, l_n) \]
\[ l_0 \in I \text{ and } l_n \in F \]
\( \dot{x} = (H(x) + u \right) \)
\(( l, a, \text{guard}_{l}, \text{jump}_{l}, l' )\)
\( l, l' \in L, a \in A \)
\( \text{guard}_{l}, \text{jump}_{l} \subset X \)
\( \text{Jump}_{l} \subset X \times X \) (a relation)

Transition from the discrete state \( l \) to \( l' \) is enabled when the continuous state \( x \in \text{guard}_{l} \), the continuous state jumps from \( x \) to \( x' \), i.e. \( (x, x') \in \text{Jump}_{l} \).

- Act: a mapping assigning to each \( l \in L \) a set of differential-algebraic equations \( F_{l} \)
  \[ F_{l}(x, \dot{x}, \omega) = 0 \]

Solutions of these equations are called activities of the location.

\( \text{State of the H.A.} \) consists of discrete \( l \in L \) and continuous \( x \in X \).

\( \text{Dynamics of H.A.} \) discrete transitions and continuous evolutions

Diagram:

- \( F_{l} \) (guard, \( x \in \text{Inv}(l) \))
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
- \( \text{Jump} \)
- \( \text{guard} \)
Definition of trajectories (solutions or runs or execution) of H.A. is:

A **continuous** trajectory \((x(t), x(w))\) associated with a location \(L\) consists of \(8\) (duration of continuous trajectory),
\[ \text{pwC } w : [0, 8] \rightarrow W \]
continuous, \(\text{pwA } x : [0, 8] \rightarrow X, \omega, t \)

- \(x(t) \in \text{Inv}(L) \quad \forall t \in (0, 8)\)
- \(F_x(x(t), \dot{x}(t), w(t)) = 0 \quad \forall t \in (0, 8)\) except at
  points of discontinuity of \(w\)

trajectory of H.A. is an (infinite) sequence of continuous trajectories
\[(x_0, x_0, x_0, x_0, \ldots)\]
\[
(x_0, x_1, x_1, x_1, x_2, x_2, \ldots) \\
\vdots
\]

sets of event times
\[t_0 = t_0, \quad t_1 = t_0 + \delta_1, \quad t_2 = t_0 + \delta_1 + \delta_2, \ldots \]

the following inclusions hold
\[x_j(t_j) \in \text{Guard}_j, \delta_{j+1} \]
\[\left(x_j(t_j), x_{j+1}(t_j) \right) \in \text{Jump}_j, \delta_{j+1} \]

The \(j\)th event \(\Rightarrow\) if the sequence has an associated symbol (label) \(\alpha_j\).
Ex. Autonomous switching (Hybrid)\[ H = 1 \quad \text{Guard: } x > \Delta \]
\[ H = -1 \quad \text{Guard: } x \leq \Delta \]
\[ \dot{x} = 1 + u \quad x \leq \Delta \]
\[ \dot{x} = -1 + u \quad x > \Delta \]

Controlled switching (manual transmission)
\[ \dot{x}_1 = x_2 \]
\[ \dot{x}_2 = -ax_2 + u \]
\[ v \in \{1, 2, 3, 4\} \quad \text{gear shift} \]
\[ a : \text{parameter} \]
\[ u : \text{input} \]

Autonomous Jump (Bouncing Ball)
\[ q > 0, \quad \dot{q} = -1 \]
\[ q = 0, \quad \dot{q}^+ = -e \dot{q}^- \]
\[ e : \text{coeff. of restitution} \]

Controlled Jump (Inventory Management)
\[ x(t): \text{amount of inventory} \]
\[ \text{Condition: } \dot{x}(t) = -Rx(t) \]
during restocking (maybe when inventory reaches a lower threshold)
\[ x(t^+) = x(t^-) + \uparrow \]
restocking amount
Features of H. dynamics.

Steps of H. evolution:

\[ l_1 \xrightarrow{\text{switch}} l_2 \text{ (can be seen as jumping)} \]

continuous trajectory

\( x \in \text{Inv}(x) \)

Guard condition

"location" (enabling conditions)

"invariant" (enforcing condition)

externally induced: "jump"

internally: "autonomous switch" "jump"

Issues:

1. System is in a state where no continuation is possible: "deadlock".

2. The sequence of \( \delta_i \) may get smaller and smaller, so that \( \sum_{i=0}^{\infty} \delta_i = 1 \) (\( 0 < \delta \)). Let \( \delta \) be an accumulation point for \( \delta \) and is an accumulation point for \( 1 \), where \( 1 \) is the maximum, i.e., bounding ball.

3. It is event: "zero behavior", e.g., bouncing ball.

4. Using limits, we start at \( 1 \).

5. Multiple event time (collisions) multiple variable behavior is more complicated.
4) Chattering: switch followed by a state where

sliding "location invariance" not satisfied
(map control)

Pulse for solution

5) Well-posed: might require a deterministic h.a.

(well-posed), i.e. having a unique solution

Trajectory

Ex. of unique

\[ \dot{y} = u \]

\[ u = +1, \quad y > 0 \]

\[ u = -1, \quad y < 0 \]

\[ -1 \leq u \leq 1, \quad y = 0 \]

- 1) \( y(t) = t \) \( (u(t) = 1) \)

2) \( y(t) = -t \) \( (u(t) = -1) \)

3) \( y(t) = 0 \) \( (u(t) = 0) \)

6) "Finite Escape Time"

\[ \dot{x}(t) = 1 + x^2(t), \quad x(0) = 0 \]

\[ \text{Soln:} \quad x(t) = \tan t \]

\[ \therefore t = \frac{\pi}{2}, \quad \text{explosion occurs} \]

7) Problematic algebraic constraint: that can be

solved using "jumps",

Remark: many other model forms

(added, stochastic H.S.)