02_Modelling Framework of Hybrid Systems

1. Frameworks

Piecewise Affine Systems (PWA)

> region partition and piece-wise affine constraint

$$\begin{aligned} x(k+1) &= A_i x(k) + B_i u(k) + f_i \\ y(k) &= C_i x(k) + D_i u(k) + g_i \end{aligned} \text{ for } \begin{bmatrix} x(k) \\ u(k) \end{bmatrix} \in \Omega_i, i = 1, \ldots, N$$

with

$\Omega_1, \cdots, \Omega_N$: **convex** polyhedra (i.e., given by finite number of linear inequalities) in input/state space, **non-overlapping** interiors

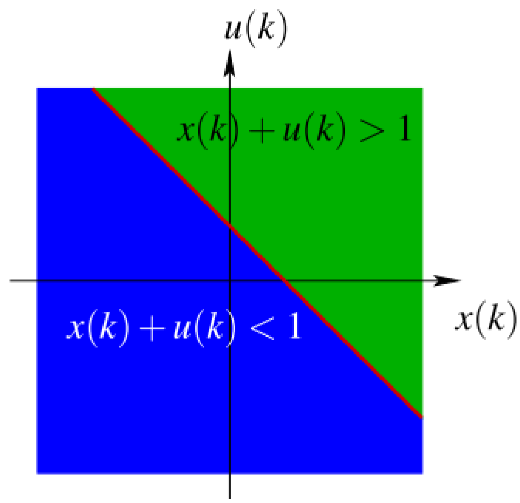
- PWA can be used as **approximation of nonlinear model:** e.g. using least square criterion

Example

Integrator with upper saturation:

$$x(k+1) = \begin{cases} x(k)+u(k) & \text{if } x(k)+u(k) \leqslant 1 \\ 1 & \text{if } x(k)+u(k) \geqslant 1 \end{cases}$$
$$y(k) = x(k)$$

deterministic?
* if in blue, no problem
* if in green, no problem
* if in red line, output is equal to 1
so, if from input-output behavior:
deterministic
if from mode perspective,
cannot deterministic



hs_mod.5

Mixed Logical Dynamical (MLD) Systems

> boolean variable + linear equality constraint

$$\begin{aligned} x(k+1) &= Ax(k) + B_1u(k) + B_2\delta(k) + B_3z(k) \\ y(k) &= Cx(k) + D_1u(k) + D_2\delta(k) + D_3z(k) \\ E_1x(k) + E_2u(k) + E_3\delta(k) &+ E_4z(k) \leqslant g_5, \end{aligned}$$

- $x(k) = \left[x_{\mathrm{r}}^{\top}(k) x_{\mathrm{b}}^{\top}(k)\right]^{\top}$ with $x_{\mathrm{r}}(k)$ real-valued, $x_{\mathrm{b}}(k)$ boolean
- $z(k)$: real-valued auxiliary variables
- $\delta(k)$: boolean auxiliary variables

Transformation of Logical part

- Associate with literal X_i logical variable $\delta_i \in \{0,1\}$:

  $\delta_i = 1$ iff $X_i = \mathrm{T}, \delta_i = 0$ iff $X_i = \mathrm{F}$

- Other Logical Operations

$$\begin{aligned} &X_1 \wedge X_2 \text{ equivalent to } \delta_1 = \delta_2 = 1 \\ &X_1 \vee X_2 \text{ equivalent to } \delta_1 + \delta_2 \geqslant 1 \\ &\sim X_1 \text{ equivalent to } \delta_1 = 0 \\ &X_1 \Rightarrow X_2 \text{ equivalent to } \delta_1 - \delta_2 \leqslant 0 \\ &X_1 \Leftrightarrow X_2 \text{ equivalent to } \delta_1 - \delta_2 = 0 \\ &X_1 \oplus X_2 \text{ equivalent to } \delta_1 + \delta_2 = 1 \end{aligned}$$

Transformation of Real-Value Function

- For $f : \mathbb{R}^n \to \mathbb{R}$ and $x \in \mathscr{X} with \mathscr{X}$ bounded, define
- $M \stackrel{\text{def}}{=} \max_{x \in \mathscr{X}} f(x) \quad m \stackrel{\text{def}}{=} \min_{x \in \mathscr{X}} f(x)$
- Mixture Logic of real-value function and boolean variables

$$
\begin{aligned}
&[f(x) \leqslant 0] \wedge [\delta = 1] \text{ true iff } f(x) - \delta \leqslant -1 + m(1 - \delta) \\
&[f(x) \leqslant 0] \vee [\delta = 1] \text{ true iff } f(x) \leqslant M\delta \\
&\sim [f(x) \leqslant 0] \text{ true iff } f(x) \geqslant \varepsilon \quad \text{(with } \varepsilon \text{ machine precision)} \\
&[f(x) \leqslant 0] \Rightarrow [\delta = 1] \text{ true iff } f(x) \geqslant \varepsilon + (m - \varepsilon)\delta \\
&[f(x) \leqslant 0] \Leftrightarrow [\delta = 1] \text{ true iff } \begin{cases} f(x) \leqslant M(1 - \delta) \\ f(x) \geqslant \varepsilon + (m - \varepsilon)\delta \end{cases}
\end{aligned}
$$

Transformation of Product of Logic Variable

Product $\delta_1 \delta_2$ can be replaced by auxiliary variable $\delta_3 = \delta_1 \delta_2$

$$
\delta_3 = \delta_1 \delta_2 \quad \text{is equivalent to} \begin{cases} -\delta_1 + \delta_3 \leqslant 0 \\ -\delta_2 + \delta_3 \leqslant 0 \\ \delta_1 + \delta_2 - \delta_3 \leqslant 1 \end{cases}
$$

Transformation of Product of Logic Variable and Real-value Function

$\delta f(x)$ can be replaced by auxiliary real variable $y = \delta f(x)$

$$
y = \delta f(x) \quad \text{is equivalent to} \begin{cases} y \leqslant M\delta \\ y \geqslant m\delta \\ y \leqslant f(x) - m(1 - \delta) \\ y \geqslant f(x) - M(1 - \delta) \end{cases}
$$

One example of transformation from PWA to MLD

- Consider PWA system:

$$x(k+1) = \begin{cases} 0.8x(k) + u(k) & \text{if } x(k) \geqslant 0 \\ -0.8x(k) + u(k) & \text{if } x(k) < 0 \end{cases}$$

where $x(k) \in [-10, 10]$ and $u(k) \in [-1, 1]$

- Associate binary variable $\delta(k)$ to condition $x(k) \geqslant 0$ such that $[\delta(k) = 1] \Leftrightarrow [x(k) \geqslant 0]$ or

$$\begin{aligned} -m\delta(k) &\leqslant x(k) - m \\ -(M+\varepsilon)\delta(k) &\leqslant -x(k) - \varepsilon \end{aligned}$$

where $M = -m = 10$, and ε is machine precision

- PWA system can be rewritten as

$$x(k+1) = 1.6\,\delta(k)x(k) - 0.8\,x(k) + u(k)$$

hs_mod.11

- $x(k+1) = 1.6\,\delta(k)x(k) - 0.8\,x(k) + u(k)$
- Define new variable $z(k) = \delta(k)x(k)$ or

$$\begin{aligned} z(k) &\leqslant M\delta(k) \\ z(k) &\geqslant m\delta(k) \\ z(k) &\leqslant x(k) - m(1-\delta(k)) \\ z(k) &\geqslant x(k) - M(1-\delta(k)) \end{aligned}$$

- PWA system now becomes

$$x(k+1) = 1.6z(k) - 0.8x(k) + u(k)$$

subject to linear constraints above $\rightarrow$ MLD

Linear Complementarity (LC) systems

> complementarity condition + linear equality constraint

$$
\begin{aligned}
x(k+1) &= Ax(k) + B_1 u(k) + B_2 w(k) \\
y(k) &= Cx(k) + D_1 u(k) + D_2 w(k) \\
v(k) &= E_1 x(k) + E_2 u(k) + E_3 w(k) + e_4 \\
0 \leqslant v(k) &\perp w(k) \geqslant 0
\end{aligned}
$$

$v(k), w(k)$: “complementarity variables” (real-valued)

- LC systems do not have δ and inequality constraints
- LC systems only have equation and complementary conditions

Extended Linear Complementarity (ELC) systems

> group complementarity condition + linear equality constraint

$$
\begin{aligned}
&x(k+1) = Ax(k) + B_1 u(k) + B_2 d(k) \\
&y(k) = Cx(k) + D_1 u(k) + D_2 d(k) \\
&E_1 x(k) + E_2 u(k) + E_3 d(k) \leqslant e_4 \\
&\textstyle\sum_{i=1}^{p} \prod_{j \in \phi_i} \left(e_4 - E_1 x(k) - E_2 u(k) - E_3 d(k)\right)_j = 0
\end{aligned}
$$

- $d(k)$: real-valued auxiliary variable
- The 4th condition is equal to $\prod_{j \in \phi_i} \left(e_4 - E_1 x(k) - E_2 u(k) - E_3 d(k)\right)_j = 0 \quad \text{for each } i \in \{1, \ldots, p\}$
  - system of linear inequalities with p groups, in each group **at least one inequality should hold with equality**

Max-Min-Plus-Scaling (MMPS) systems

Max-min-plus-scaling expression :

$$
f := x_i |\alpha| \max\left(f_k, f_l\right) |\min\left(f_k, f_l\right)| \, f_k + f_l \mid \beta f_k
$$

with $\alpha, \beta \in \mathbb{R}$ and f_k, f_l again **MMPS expressions.**

MMPS Systems

$$
\begin{aligned}
&x(k+1) = \mathscr{M}_x(x(k), u(k), d(k)) \\
&y(k) = \mathscr{M}_y(x(k), u(k), d(k)) \\
&\mathscr{M}_c(x(k), u(k), d(k)) \leqslant c \\
&\text{ with } \mathscr{M}_x, \mathscr{M}_y, \mathscr{M}_\mathrm{c} \text{ MMPS expressions}
\end{aligned}
$$

$d(k)$: real-valued auxiliary variables

- It is good for systems with soft & hard synchronization constraints

Some Illustrated Example

- **max**: one of two product is ready is okay
- **min**: both of two product are needed

Example

- Integrator with upper saturation:

$$x(k+1) = \begin{cases} x(k)+u(k) & \text{if } x(k)+u(k) \leqslant 1 \\ 1 & \text{if } x(k)+u(k) \geqslant 1 \end{cases}$$
$$y(k) = x(k)$$

can be recast as

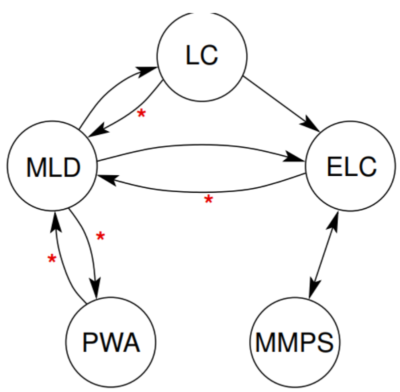
$$x(k+1) = \min(x(k)+u(k), 1)$$
$$y(k) = x(k)$$

2. Equivalence of MLD, LC, ELC, PWA, and MMPS systems

Definition of Equivalence

Equivalence between model classes $\mathscr{A}$ and $\mathscr{B}$:
for each model $\in \mathscr{A}$ there exists model $\in \mathscr{B}$ with same i**nput/output behavior** (+ vice versa)

Equivalence Relations



- Red Star means this transformation need some constraints

Different Advantages

Each subclass has own advantages:

- **stability criteria** for PWA
- control and **verification** techniques for MLD
- **control** techniques for MMPS

- conditions of **existence and uniqueness** of solutions for LC
- **transfer techniques** from one class to other

Based on the transformation, we can transform among them for different usage.

3. Transformation Among Different Models

MLD and LC systems

Theorem

Every MLD system can be written as LC system

Method

- $\delta_i(k) \in \{0,1\}$ is equivalent to $0 \leqslant \delta_i(k) \perp 1-\delta_i(k) \geqslant 0$
  $\rightarrow$ introduce auxiliary variable $p(k) = [1\ \ 1\ \dots\ 1]^\top - \delta(k)$ with

$$0 \leqslant \delta(k) \perp p(k) \geqslant 0$$

- For constraint $E_1x(k)+E_2u(k)+E_3\delta(k)+E_4z(k) \leqslant g_5$, introduce auxiliary variables $q(k) = g_5 - E_1x(k) - E_2u(k) - E_3\delta(k) - E_4z(k) \geqslant 0$ and $r(k) \geqslant 0$ with

$$0 \leqslant q(k) \perp r(k) \geqslant 0$$

  Complementarity condition will always hold (select $r(k)=0$)

- For LC: all variables $\geqslant 0$
  $\rightarrow$ split real-valued variable $z(k)$ in "positive" and "negative part":
  $z(k) = z^+(k) - z^-(k)$ with $z^+(k) = \max(0, z(k))$, $z^-(k) = \max(0, -z(k))$
  or $0 \leqslant z^+(k) \perp z^-(k) \geqslant 0$

$$
\begin{aligned}
x(k+1) &= Ax(k) + B_1u(k) + \begin{bmatrix} B_2 & 0 & B_3 & -B_3 \end{bmatrix} w(k) \\
y(k) &= Cx(k) + D_1u(k) + \begin{bmatrix} D_2 & 0 & D_3 & -D_3 \end{bmatrix} w(k)
\end{aligned}
$$

$$
\underbrace{\begin{pmatrix} p(k) \\ q(k) \\ s(k) \\ t(k) \end{pmatrix}}_{=:v(k)} = \begin{pmatrix} e \\ g_5 - E_1x(k) - E_2u(k) \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -I & 0 & 0 & 0 \\ -E_3 & 0 & -E_4 & E_4 \\ 0 & 0 & 0 & I \\ 0 & 0 & I & 0 \end{pmatrix} \underbrace{\begin{pmatrix} \delta(k) \\ r(k) \\ z^+(k) \\ z^-(k) \end{pmatrix}}_{=:w(k)}
$$

$$0 \leqslant v(k) \perp w(k) \geqslant 0$$

Theorem

Every LC system can be written as MLD provided that $w(k)$ and $v(k)$ are **bounded**

Method

- Transform $v(k)$ and $w(k)$
  - LC complementarity condition $0 \leqslant v(k) \perp w(k) \geqslant 0$ implies that for each i we have $v_i(k) = 0, w_i(k) \geqslant 0$ or $v_i(k) \geqslant 0, w_i(k) = 0$
  - **Introduce boolean vector** $\delta(k)$ such that

$$
\begin{gathered}
v_i(k) = 0, w_i(k) \geqslant 0 \leftrightarrow \delta_i(k) = 1 \\
v_i(k) \geqslant 0, w_i(k) = 0 \leftrightarrow \delta_i(k) = 0
\end{gathered}
$$

  - Can be achieved by **introducing constraints**

$$
\begin{aligned}
& w(k) \leqslant M_w \delta(k) \\
& v(k) \leqslant M_v \left([11 \ldots 1]^\top - \delta(k)\right) \\
& w(k), v(k) \geqslant 0
\end{aligned}
$$

    with M_w, M_v diagonal matrices containing **upper bounds** on $w(k), v(k)$(in practice, the upper bounds usually known due to physical reasons/insight)

  - use $z(k)$ to repesent $w(k)$
  - replacing linear equality constraints: Replacing $v(k)$ by $E_1 x(k) + E_2 u(k) + E_3 w(k) + e_4$ in inequalities finally results in MLD model

$$
\begin{gathered}
x(k+1) = Ax(k) + B_1 u(k) + B_2 z(k) \\
y(k) = Cx(k) + D_1 u(k) + D_2 z(k) \\
\left[\begin{array}{c} 0 \\ E_1 \\ 0 \\ -E_1 \end{array}\right] x(k) + \left[\begin{array}{c} 0 \\ E_2 \\ 0 \\ -E_2 \end{array}\right] u(k) + \left[\begin{array}{c} -M_w \\ M_v \\ 0 \\ 0 \end{array}\right] \delta(k) + \left[\begin{array}{c} I \\ E_3 \\ -I \\ -E_3 \end{array}\right] z(k) \leqslant \left[\begin{array}{c} 0 \\ M_v e - e_4 \\ 0 \\ e_4 \end{array}\right]
\end{gathered}
$$

    with $e = \left[\begin{array}{llll} 1 & 1 & \ldots & 1 \end{array}\right]^\top, z(k) = w(k)$

LC and ELC systems

Theorem

Every LC system can be written as ELC system

$$
v(k) \perp w(k) \text{ is equivalent to } \sum v_i(k) w_i(k) = 0
$$

PWA and MLD systems

Theorem

Well-posed PWA system can be rewritten as MLD system assuming that set of feasible states and inputs is bounded

See the example

Theorem

Completely well-posed MLD can be rewritten as PWA

well-posed: given $x(k), u(k)$, then $x(k+1), u(k+1)$ are unique

completely well-posed: given $x(k), u(k)$, then $x(k+1), u(k+1), \delta(k), z(k)$ are unique. In MLD it means the inequalities can only yield one possible solution

MMPS and ELC

Theorem

The classes of MMPS and ELC systems **coincide**

MMPS $\subseteq$ ELC

- Expressions of form $f = x_i, f = \alpha, f = f_k + f_l, f = \beta f_k$ result in linear equations
- $f = \max(f_k, f_l) = -\min(-f_k, -f_l)$ can be rewritten as

$$f - f_k \geqslant 0, \quad f - f_l \geqslant 0, \quad (f - f_k)(f - f_l) = 0$$

- Two or more ELC systems can be combined into one large ELC

ELC $\subseteq$ MMPS

- Linear equations are MMPS expressions
- Complementarity condition can be transformed
  - $\forall i, \exists j \in \phi_i \text{ such that } \underbrace{(e_4 - E_1 x(k) - E_2 u(k) - E_3 d(k))_j}_{\geqslant 0} = 0$
  - Then: $\min_{j \in \phi_i} (e_4 - E_1 x(k) - E_2 u(k) - E_3 d(k))_j = 0 \quad \text{for each } i$

MLD and ELC systems

Theorem

Every MLD system can be rewritten as ELC system

- boolean variables:

$$\begin{aligned} &-\delta_i(k) \leqslant 0 \\ &\delta_i(k) \leqslant 1 \\ &\delta_i(k)(1 - \delta_i(k)) = 0 \end{aligned}$$

- Note, if want to direct transform MLD to MMPS:

$$\begin{aligned} &\max(-\delta_i(k), \delta_i(k) - 1) = 0 \\ &\min(\delta_i(k), 1 - \delta_i(k)) = 0 \end{aligned}$$

Theorem

Every ELC system can be written as MLD system, provided that $e_4 - E_1 x(k) - E_2 u(k) - E_3 d(k)$ is **bounded**

- Transform Complementary Condition

$$\begin{aligned}&(e_4)_j - (E_1 x(k) + E_2 u(k) + E_3 d(k))_j \leqslant M_j \delta_j(k) \quad \text{for each } j \in \phi_i \\ &\textstyle\sum_{j \in \phi_i} \delta_j(k) \leqslant \# \phi_i - 1\end{aligned}$$

with $\delta_j(k) \in \{0, 1\}$ auxiliary variables, and M_j upper bound for $(e_4 - E_1 x(k) - E_2 u(k) - E_3 d(k))_j$

An example

See Lecture

4. Timed Automata

Definition: Rectangular Sets

Subset of $\mathbb{R}^n$ set is called **rectangular** if it can be written as finite boolean combination of constraints of form

$$x_i \leqslant a, \quad x_i < b, \quad x_i = c, \quad x_i \geqslant d, \quad x_i > e$$

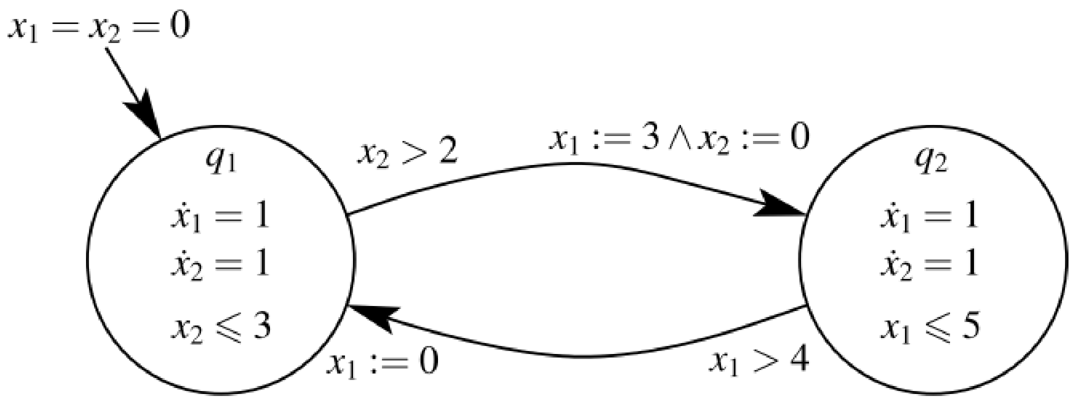
- Rectangular sets are "rectangles" or "boxes" in $\mathbb{R}^n$ whose sides are **aligned with the axes**, or **unions** of such rectangles/boxes(including empty set)

Timed Automaton

Timed automaton is hybrid automaton with following characteristics:

- automaton involves **differential equations of form** $\dot{x}_i = 1$ continuous variables governed by this differential equation are called **"clocks" or "timers"**
- sets involved in definition of initial states, guards, and invariants are **rectangular sets**
- reset maps involve either **rectangular set**, or may **leave certain states unchanged**

Properties



Timed automata involve simple **continuous dynamics**

- all differential equations of form $\dot{x} = 1$ (**clock dynamics**)
- all invariants, guards, etc. involve comparison of real-valued states **with constants**
- reset maps involve either **rectangular set,** or may leave certain states **unchanged**

Timed automata are **limited for modeling physical system**s and very well suited for **encoding timing constraints**

5. Timed Petri Nets

- **Transition enabled** if all input places $(\bullet_t)$ contain at least 1 token

Compared to Untimed Petri Nets, Timed Petri Nets has **two more variables**:

- **discrete state variables** (markings, $m_\theta(p)$)
- **continuous state variables**(arrival times $M_\theta(p)$)

$M_\theta(p) := \{\theta_1, \ldots, \theta_{m_\theta(p)}\}$, with arrival times $\theta_1 \leqslant \theta_2 \leqslant \ldots \leqslant \theta_{m_\theta(p)}$ of $m_\theta(p)$ token in place p

And we also have **interval** $[L(t), U(t)]$: time interval for specified token during which it must be transmitted

Time Analysis

Transition t becomes enabled at

$$\max_{p\in\bullet} \min M_\theta(p)$$

Then transition t may fire at some time

$$\theta \in \left[\max_{p\in^\bullet t} \min M_\theta(p) + L(t), \max_{p\in^\bullet t} \min M_\theta(p) + U(t)\right]$$

- If enabling condition is still **valid at final time** of firing interval, then transition is **forced to fire**
- However, many problems are **undecidable or NP-hard**

Summary

- Frameworks
  - PWA
  - MLD
  - LC
  - ELC
  - MMPS
- Equivalence of these systems
- Transformation among these models
- Timed automata and Timed Petri Nets