Efficient Mode Enumeration of Compositional Hybrid Models

F. D. Torrisi, T. Geyer, M. Morari

torrisi|geyer|morari@aut.ee.ethz.ch, http://control.ethz.ch/~hybrid

Automatic Control Laboratory

Swiss Federal Institute of Technology (ETH)

Thanks to Komei Fukuda and Mato Baotic

Modeling Requirements

Linear dynamics changes according to

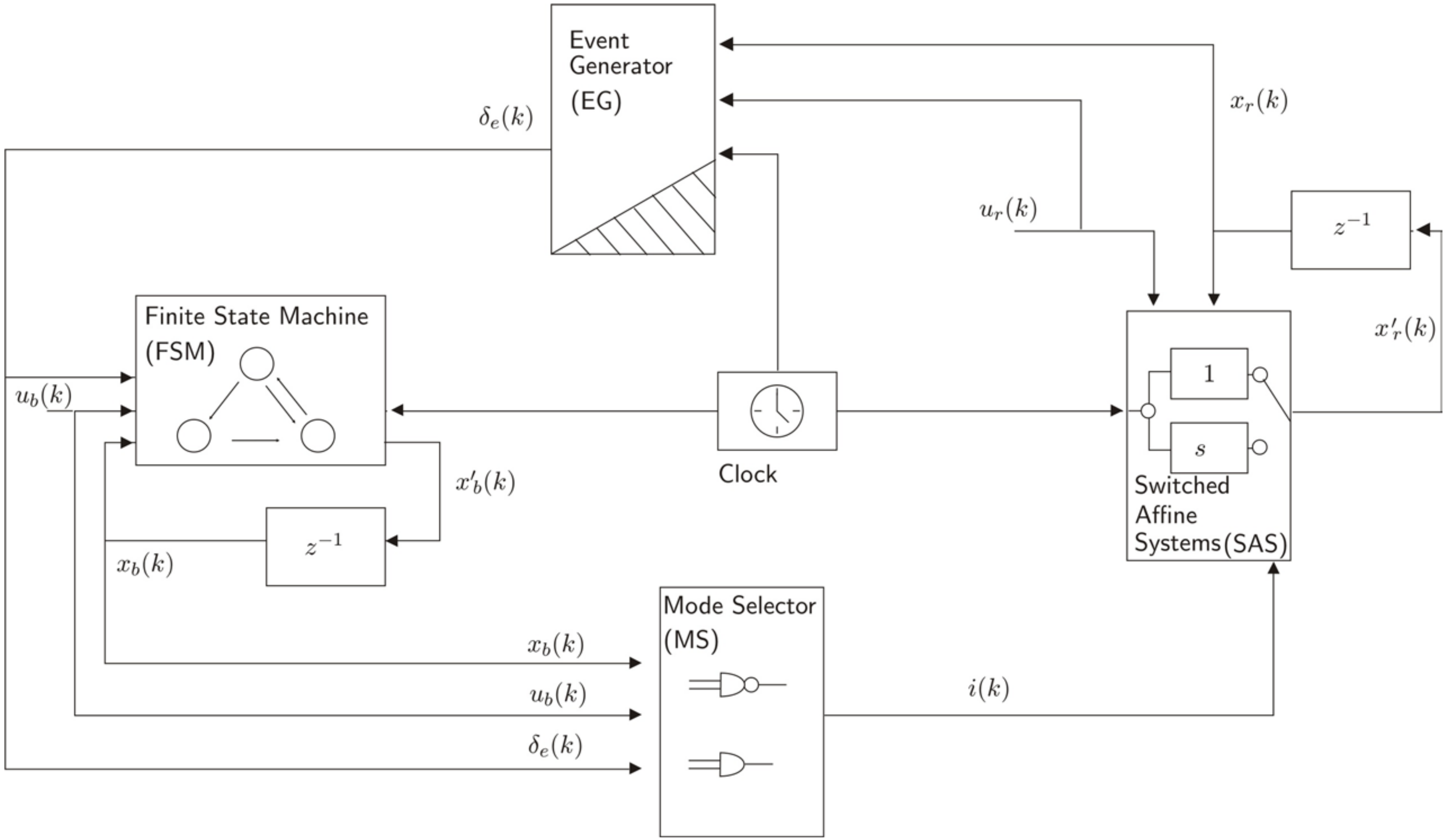
- Logic state
- Exogenous logic inputs
- Threshold conditions
- Time
- Any logic combination of the former

Example:

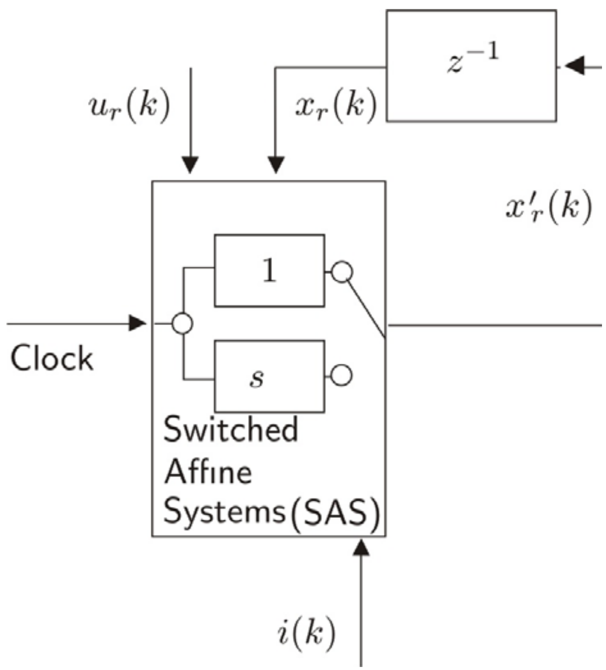
$$\text{IF } ((t > T_1) \vee (x_1 \leq 7)) \wedge (u_1 = \text{TRUE})$$
$$\text{THEN } x(t+1) = A_j x(t) + B_j u(t) + f_j$$

ETH ifa

Discrete Hybrid Automata



Switched Affine Systems

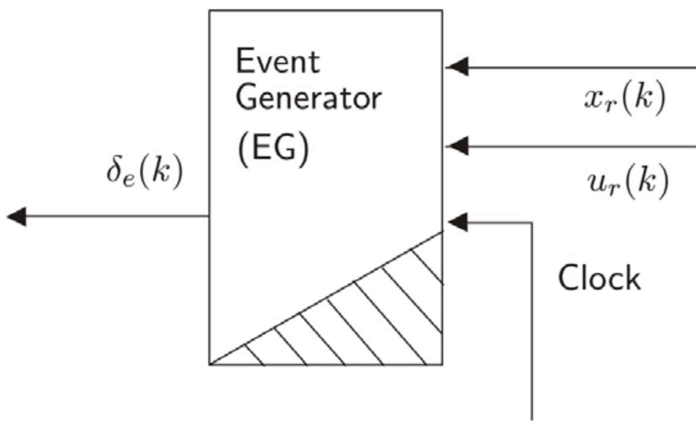


Linear affine dynamics depends upon the mode selector $i(t)$

$$
\begin{aligned}
x'_r(k) &= A_{i(k)} x_r(k) + \\
&\quad B_{i(k)} u_r(k) + f_{i(k)}
\end{aligned}
$$

$$
\begin{aligned}
y_r(k) &= C_{i(k)} x_r(k) + \\
&\quad D_{i(k)} u_r(k) + g_{i(k)}
\end{aligned}
$$

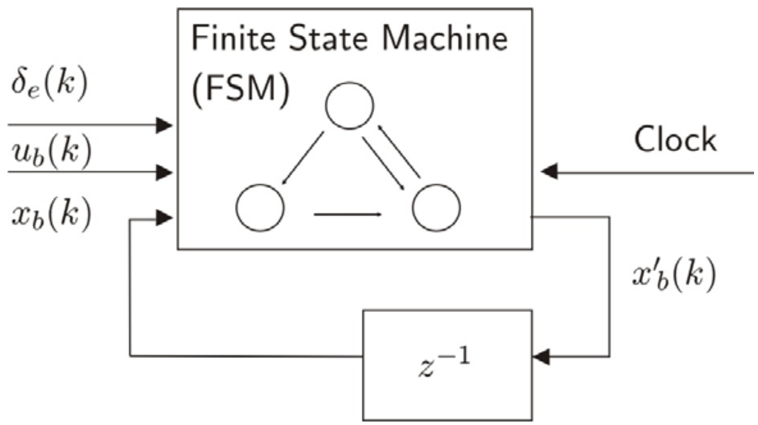
Event Generator



Generates a logic signal according to the satisfaction of a linear affine constraint

$$\delta_e(k) = f_{\mathsf{H}}(x_r(k), u_r(k), k)$$

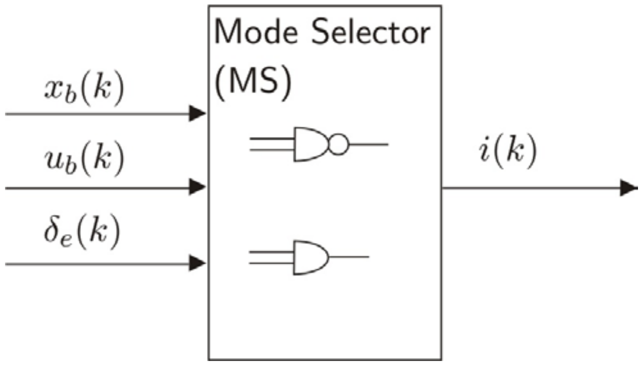
Finite State Machine



Discrete dynamic process
Evolves according to a logic state update function

$$
\begin{aligned}
x'_b(k) &= f_{\mathsf{B}}(x_b(k), u_b(k), \delta_e(k)) \\
y_b(k) &= g_{\mathsf{B}}(x_b(k), u_b(k), \delta_e(k))
\end{aligned}
$$

Mode Selector

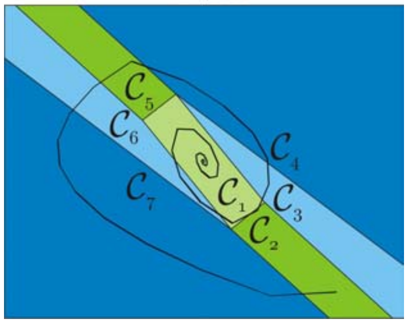


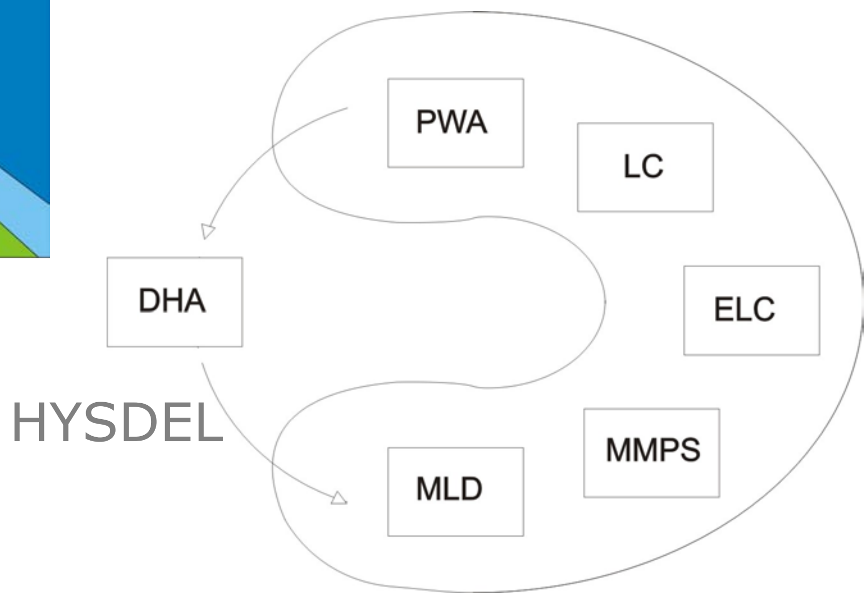
A Boolean function selects the active mode $i(k)$ of the SAS

$$i(k) = f_{\mathsf{M}}(x_b(k), u_b(k), \delta_e(k))$$

DHA and Other Modeling Frameworks

Piecewise Affine Models (PWA) define an affine dynamics on each cell of a polyhedral partition

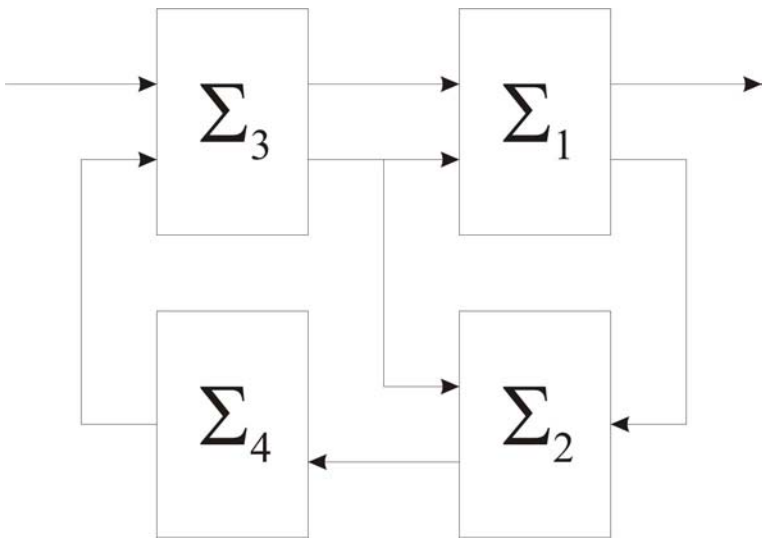




Mixed Logic Dynamical Models (MLD) are
linear systems
plus mixed integer
inequalities

$$
\begin{aligned}
x'(k) &= Ax(k) + B_1u(k) + B_2\delta(k) + B_3z(k) + B_5 \\
y(k) &= Cx(k) + D_1u(k) + D_2\delta(k) + D_3z(k) + D_5 \\
E_2\delta(k) + E_3z(k) &\leq E_1u(k) + E_4x(k) + E_5
\end{aligned}
$$

Composition of DHAs



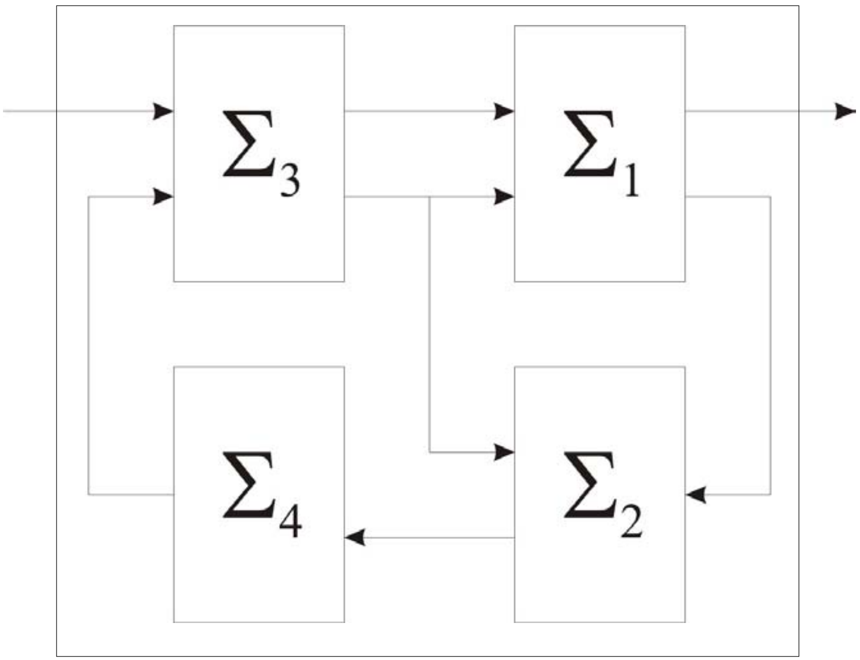
More DHAs can be combined to form a compositional DHA

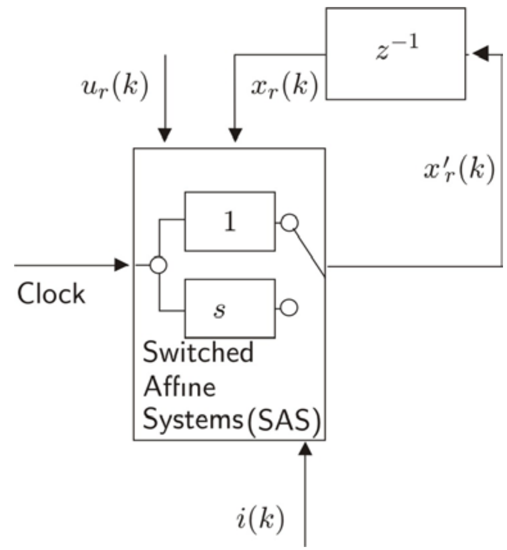
Many times hierarchical structures cannot be identified

But interaction among the parts removes modes and reduces complexity

Problem Statement

Enumerate all the possible modes of a given composition of DHAs,

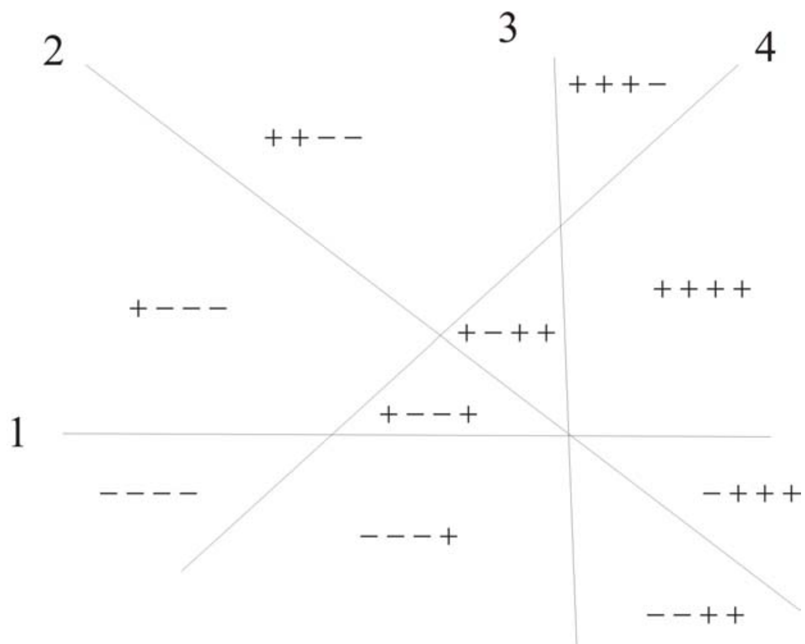




Number of possible modes could be exponential in the number of composing systems

Hyperplane Arrangements

Buck 1943, Edelsbrunner 1987, Fukuda 1996



Let $\mathcal{A}=\{H_i\}_{i=\{1,\dots,n\}}$, $H_i=\{x{:}a_ix-b_i=0\}$ be a collection of n hyperplanes in $\mathbb{R}^d$

Theorem Each polyhedral region (or cell) is associated to a sign marking

Theorem The total number of cells is bounded by Buck's formula

$$\#M \leq \sum_{i=0}^{d} \binom{n}{i}$$

Hyperplane Arrangements - Algorithms

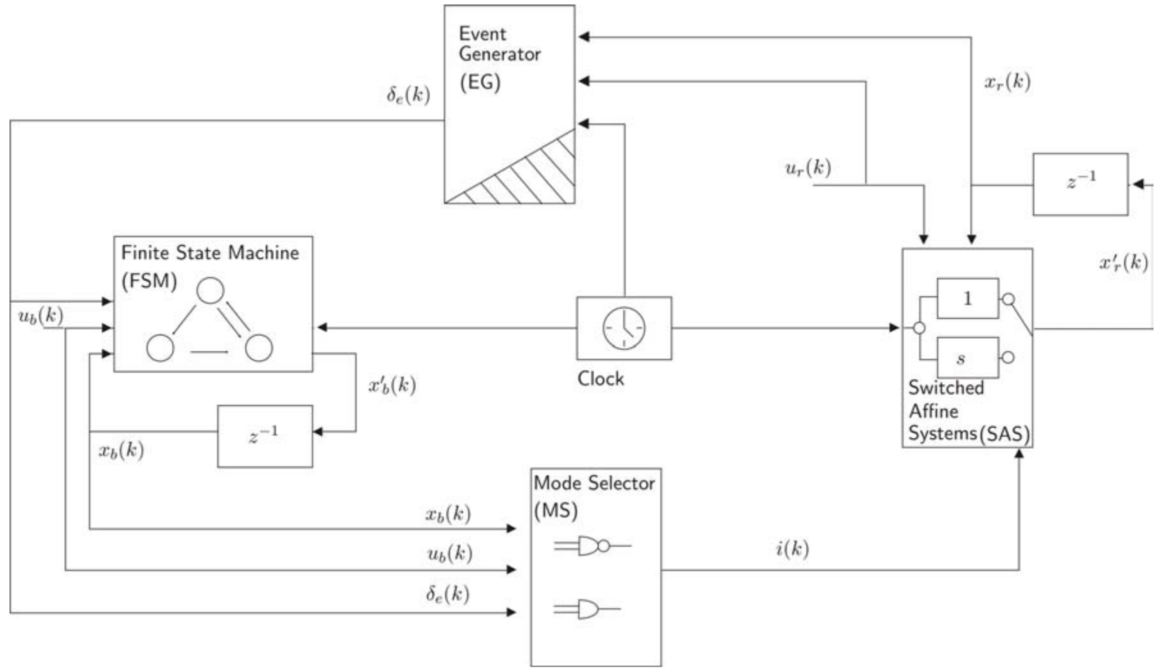
There is an optimal algorithm for enumeration of hyperplane arrangements with time and space complexity $O(n^d)$ (Edelsbrunner '87)

There is reverse search algorithm (Fukuda '96,'01) for enumeration of hyperplane arrangements that runs in $O(n\ \mathrm{lp}(n,d)\ \#M)$ time and $O(n,d)$ space, where $\mathrm{lp}(n,d)$ is the complexity of solving a linear program with d variables and n constraints

Let $M=\mathrm{hyparr}(\mathcal{A},\mathcal{R})$ be a function that computes all the sign markings of the hyperplane arrangement $\mathcal{A}$ in region $\mathcal{R}$. Let $\mathcal{P}_m$, $m \in M$ be the polyhedron associated with the sign marking m

ETH ifa

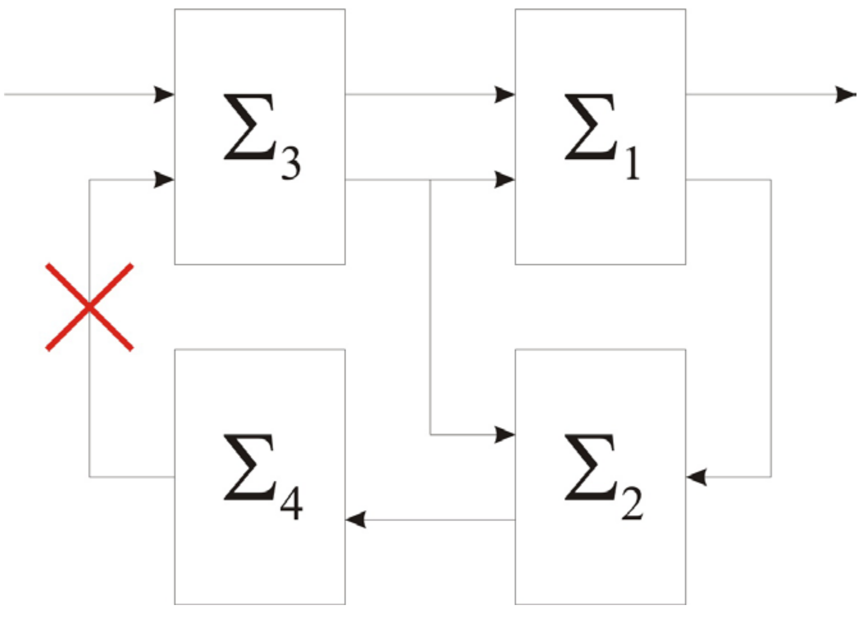
Single DHA



The Event Generator defines a hyperplane arrangement on the input+state space

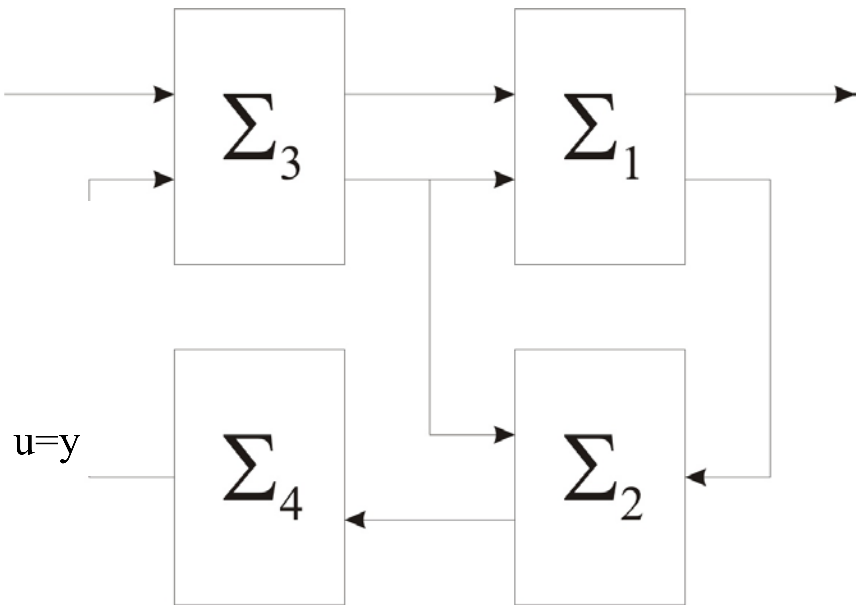
For any discrete state, discrete input and cell of the arrangement there is exactly one feasible mode

Composition of DHAs - Part 1



- Connection of DHAs is an oriented graph
- Replace the loops with constraints (u=y)
- Determine the computational order $\mathcal{O}$ by topological sorting

Composition of DHAs - Part 2



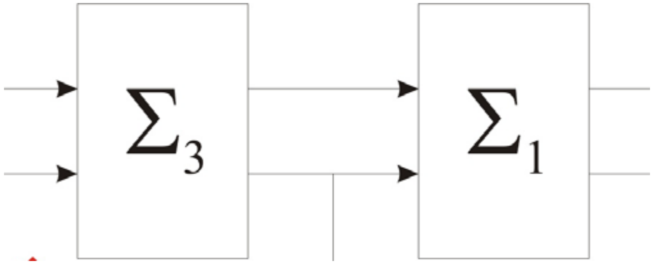
$\mathcal{O} = [3, 1, 2, 4]$

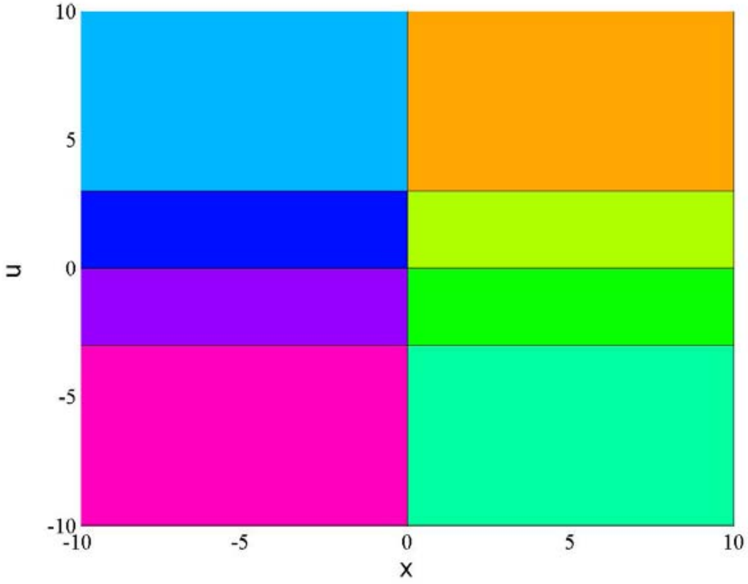
function enum($\mathcal{R}$,i),
if $\exists\ \Sigma_{O(i)}$,$\mathcal{R} \neq \emptyset$
    M=hyparr(EG$_{\mathcal{O}(i)}$,$\mathcal{R}$);
    foreach $m \in M$,
        $\mathcal{Q}$=$\mathcal{Q}$∪enum($\mathcal{P}_m$,i);
else
    if sat($\mathcal{R}$)
        return $\{\mathcal{R}\}$
    else
        return $\emptyset$

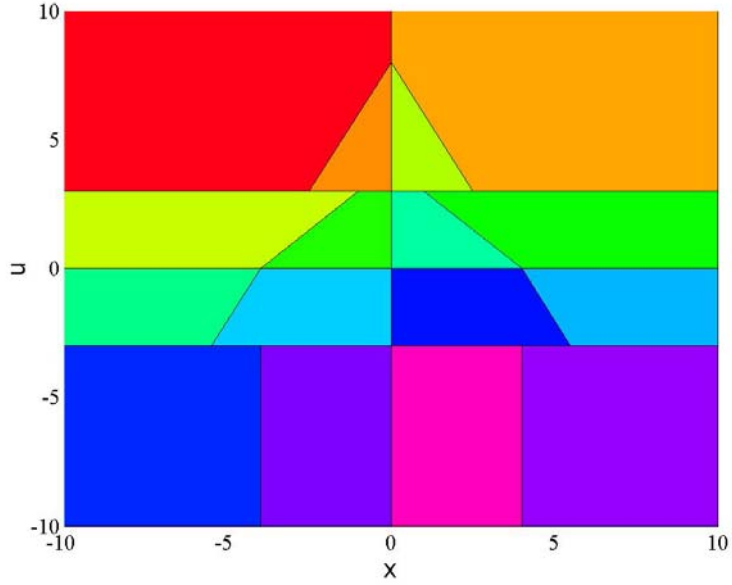
$\mathcal{Q}=\emptyset$
foreach x_b, u_b,
    $\mathcal{Q} = \mathcal{Q} \cup$ enum($\mathbb{R}^d$,1);

Composition of DHAs - Example

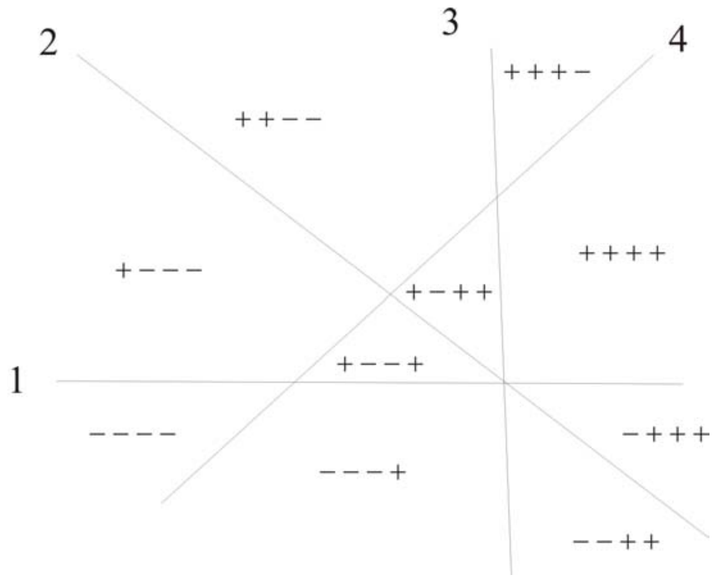
Let Σ_3 have $u \in \mathbb{R}$, $x \in \mathbb{R}$, Σ_1 be a static map

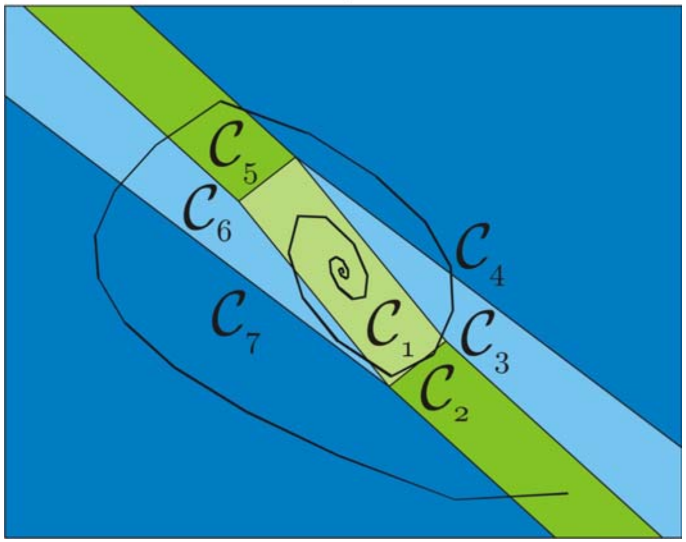






Application: Efficient PWA conversion





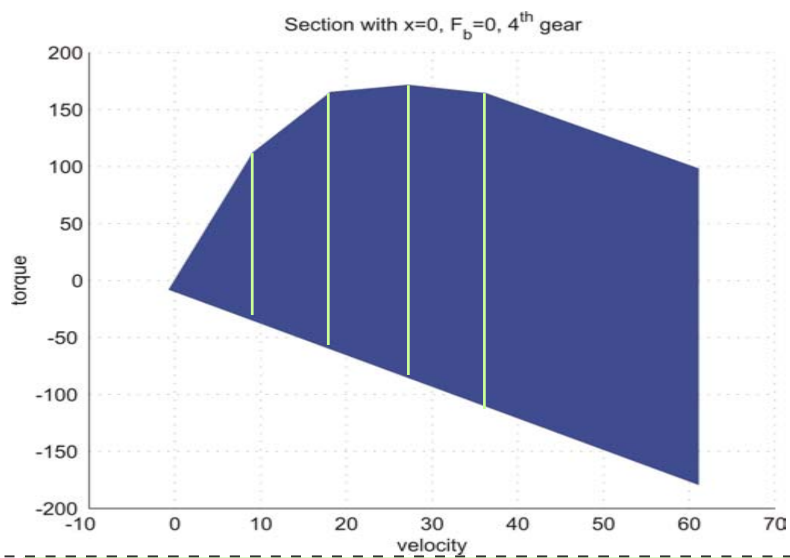
For any discrete state, discrete input and cell of the arrangement there is exactly one feasible mode, $\Rightarrow$ PWA model

Similar to Bemporad '02

Example: Car

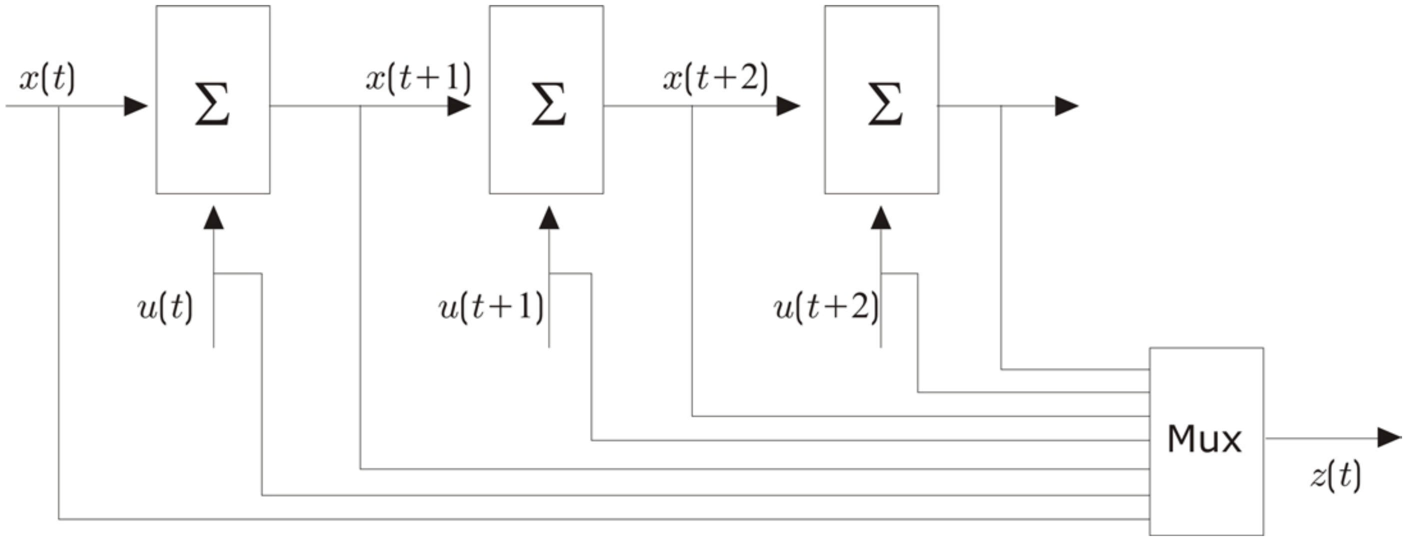
Renault Clio 1.9 DTI RXE

Continuous (gas pedal, and brakes) and discrete (gear ratio) inputs



30 regions and 6 modes enumerated in 7.5 s on PC 650MHz

Application: Optimal Control & MPC



$$\min \sum_{t<N} (|u(t)|_P + |x(t+1)|_Q) = \min |z(t)|_R$$

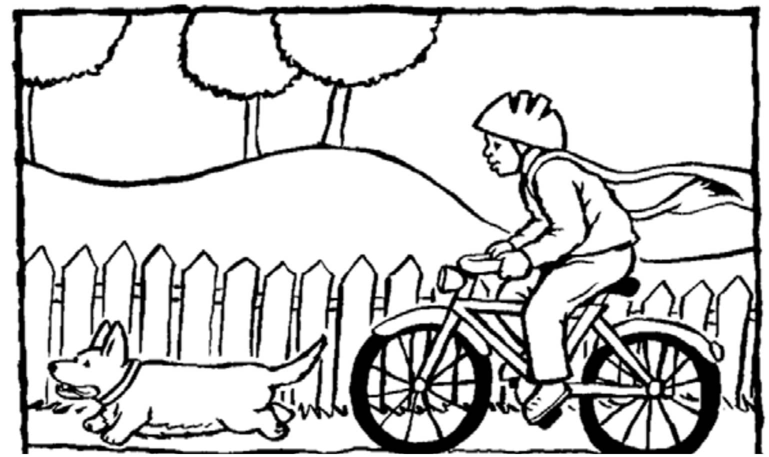
Model Predictive Control (MPC) amounts to apply optimal control in receding horizon

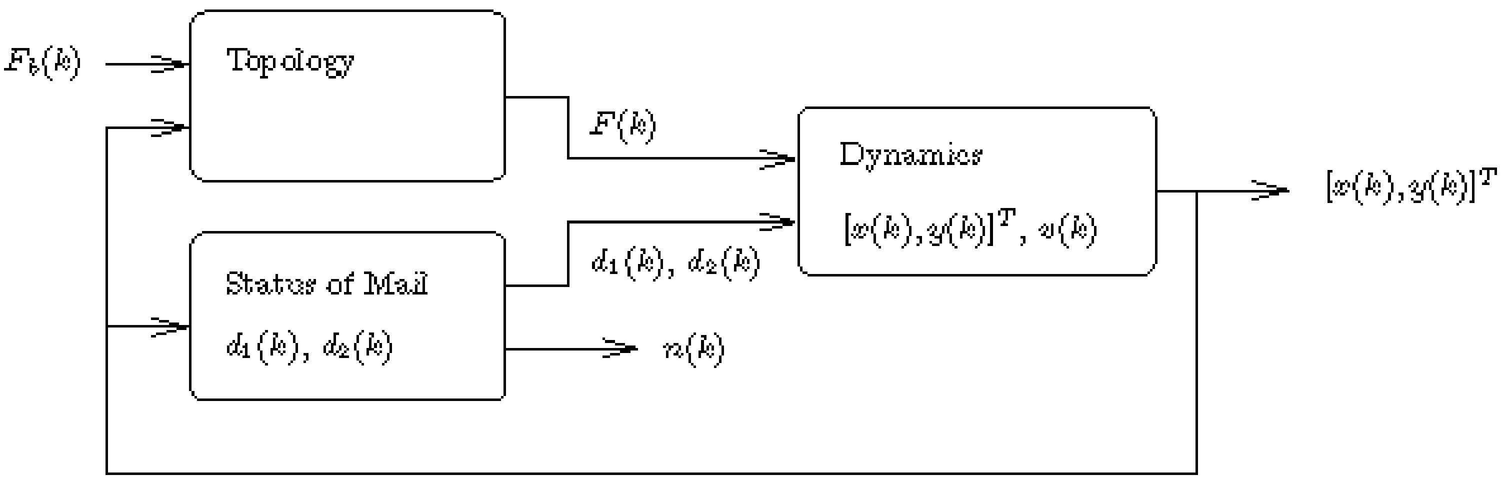
Example: Paperboy

Deliver newspapers to 2 households

Piecewise affine slope hill

Uses MPC(!!) to optimize his trajectory





Model: 3+8+0 constraints in the EGs, 4·42 cells, 36 modes

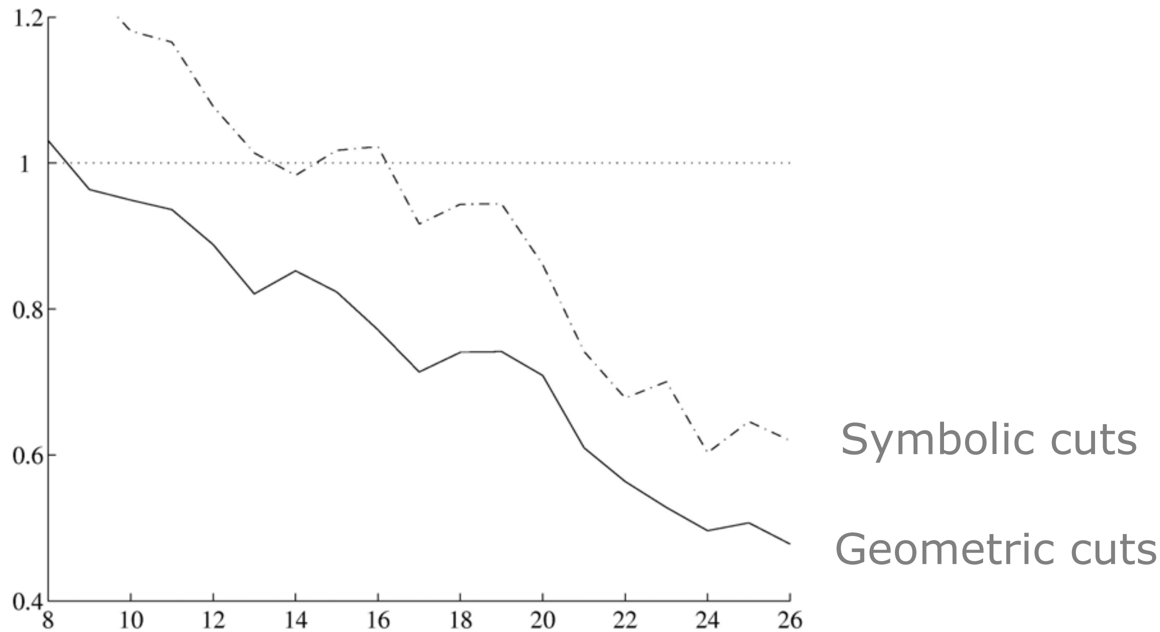
MLD model: 132 constraints, 21 integer variables

ETH ifa

Example: Paperboy

Homemade MIQP solvers allow full control of branching strategies ⇒ speedup factor 210 with prediction horizon 3

Commercial solvers (i.e. CPLEX) have less freedom, and usually include a rich bag of branching heuristics ⇒ add cuts



Normalized time vs Prediction horizon (CPLEX)

Advantage appears evident with long prediction horizonts

Conclusions

DHA models:

- Capture hybrid phenomena
- Linear dynamics
- Logic-, threshold-, and time-based switching

Cell/Region/Modes enumeration:

- A tool from computational geometry may help in the hybrid domain
- Compute an equivalent PWA model
- Reduces the complexity of MPC

Open Problems:

- Exploit sign information to merge cell with the same dynamics
- Extension to other applications (continuous time systems)