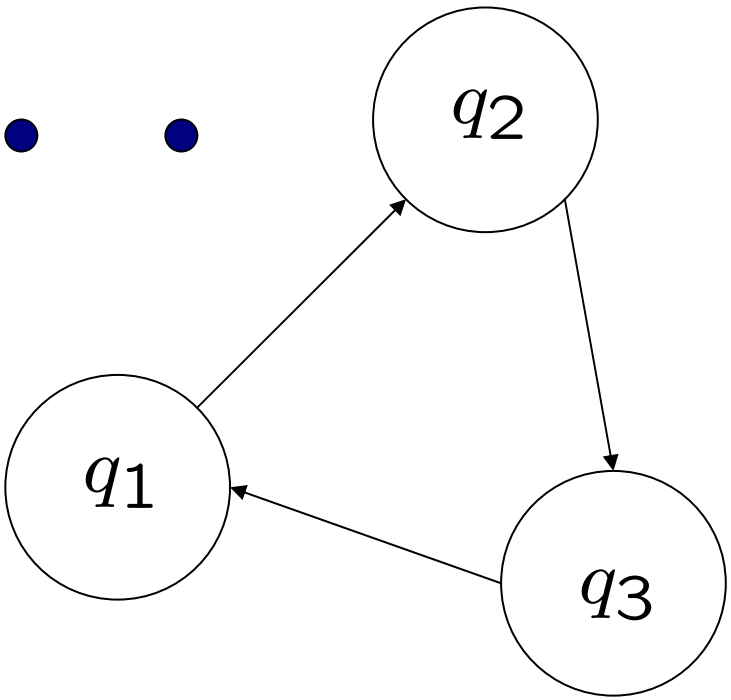
10 tools for the Simulation, Verification and Synthesis of Hybrid Systems

Alessandro Pinto
545P Cory Hall
U.C. Berkeley
apinto@eecs.berkeley.edu

Hybrid Systems

- **A Hybrid System is tuple**

$$\mathcal{H} = (\mathbf{Q}, \mathbf{U}_D, E, X, U, V, \mathcal{S}, Inv, R, G)$$



$$\mathcal{S}(q_1) = (\dot{x} = f(x, u, t))$$

$$G(q_1, q_2) = (x_1 \geq x_2 \wedge v_1 > v_2)$$

$$Inv(q_1) = (\bar{G}(q_1, q_2))$$

$$R(q_1, q_2) = (x' = r(x, u))$$

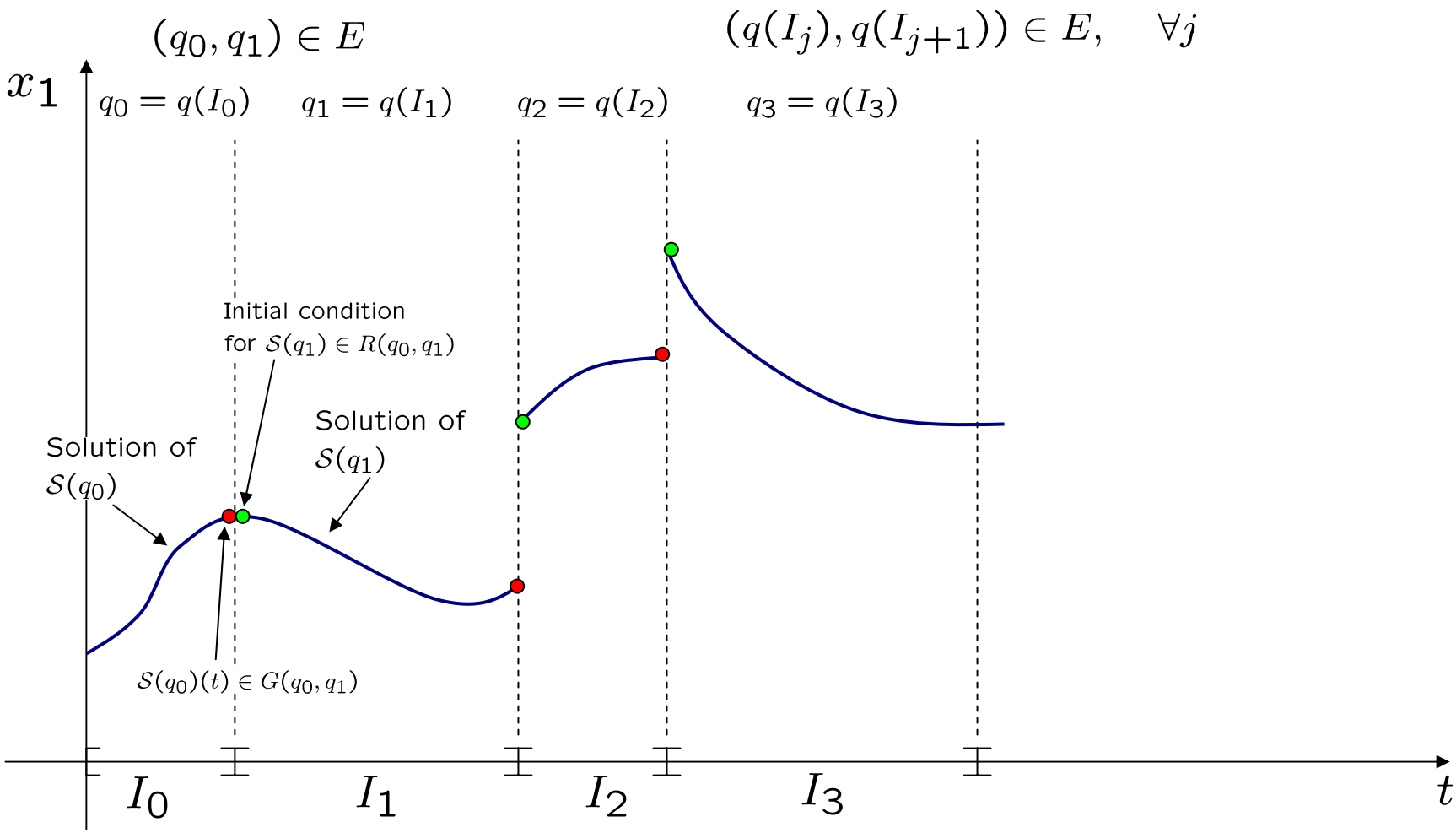
Hybrid Time Basis

A hybrid time basis τ is a finite or an infinite sequence of intervals

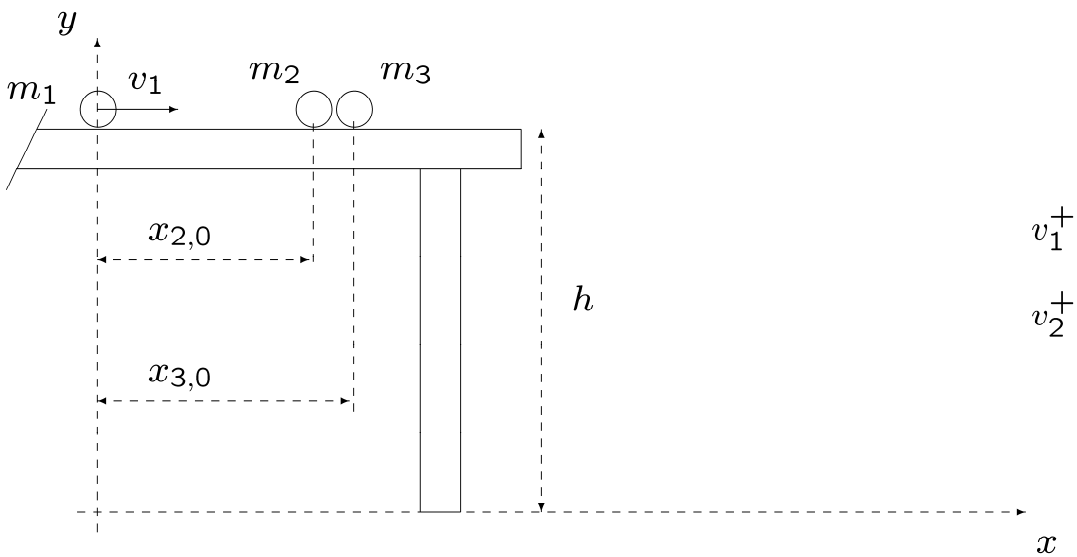
$$I_j = \{t \in \mathbb{R} : t_j \leq t \leq t'_j\},$$

where $t_j \leq t'_j$ and $t'_j = t_{j+1}$.

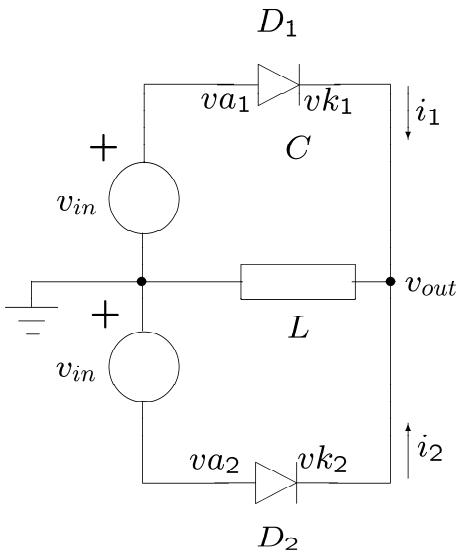
Hybrid System Execution

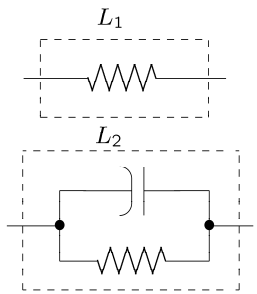


Examples



$$v_1^+ = v_1\frac{(m_1 - \epsilon m_2)}{m_1 + m_2} + v_2\frac{m_2(1+\epsilon)}{m_1 + m_2}$$
$$v_2^+ = v_1\frac{(1+\epsilon)m_1}{m_1 + m_2} + v_2\frac{(m_2 - \epsilon m_1)}{m_1 + m_2}$$





$$i = \begin{cases} \frac{v_a - v_k}{R_f} & if \quad v_a - v_k \geq 0 \\ -I_0 & otherwise \end{cases}$$

Simulink-Stateflow

- Stateflow for representing discrete dynamics
- Simulink for continuous (and discrete) dynamics
- Continuous-Discrete interaction:
  - stateflow events affecting continuous states
  - threshold detectors generating events for advancing discrete state

Simulation

The final table

Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
Simulink/Stateflow	Stateflow and Simulink switches	Stateflow output selecting state evolution	No limitations	Conditions on Stateflow inputs and threshold crossing detector	Not supported	Integrator's reset from Stateflow output

Modelica

- Non-causal modeling, object oriented
- Equivalent description of a Modelica program

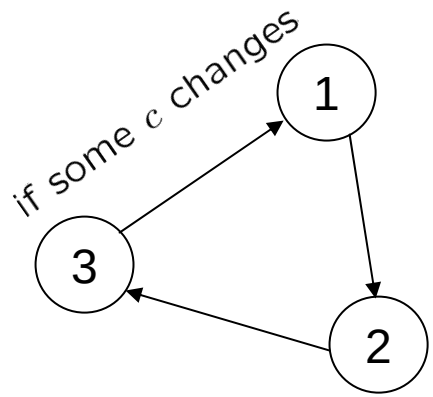
$$c := f_c(rel(v)) \quad (1)$$

$$m := f_m(v, c) \quad (2)$$

$$0 := f_x(v, c) \quad (3)$$

$v := [\dot{x}; x; y; t; m; pre(m); p]$

x	differentiated variables
y	non-differentiated variables
p	parameters
$rel(v)$	relations on variables in v
c	expressions in if statements



Simulation

The final table

Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
Simulink/Stateflow	Stateflow and Simulink switches	Stateflow output selecting state evolution	No limitations	Conditions on Stateflow inputs and threshold crossing detector	Not supported	Integrator's reset from Stateflow output
Modelica	Not explicitly defined	Events enabling equations	No limitations	Triggering relations on variables (`when` statement)	Not an explicit language feature	Through `reinit` statement

HyVisual

- **Continuously evolving signal**

  $x : T \times \mathbb{N} \rightarrow V$

- **Initial and final value functions**

  $x_i : T \rightarrow V \quad x_f : T \rightarrow V$

  $\forall t \in T, \quad x_i(t) = x(t, 0) \quad \text{and} \quad x_f(t) = x(t, m)$

Init	$t^* = t_0,\ x(t^*, 0) = x_0$
Discrete Phase	execute the model until $x_f(t^*)$ is computed;
Continuous Phase	compute t_1 such that g is continuous and locally Lipschitz on $[t^*, t_1)$. Solve $\dot{x}(t) = g(x(t), t)$ on the interval $[t^*, t_1)$ with initial condition $x_0 = x_f(t^*)$;
Iterate	Set $t^* = t_1$ and iterate from the discrete phase with $x(t^*, 0)$ equal to the value of x at t_1 computed in the previous step.

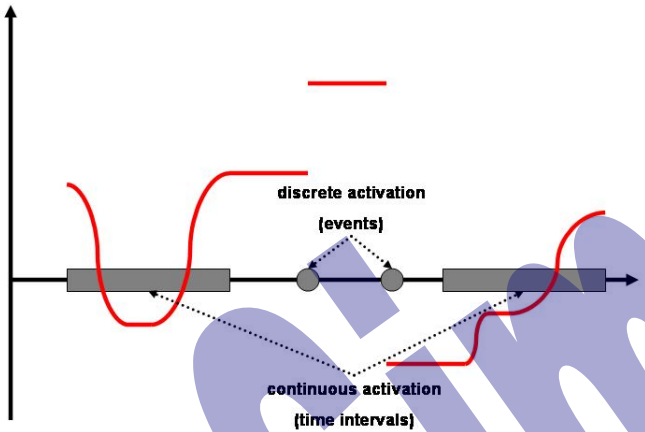
Simulation

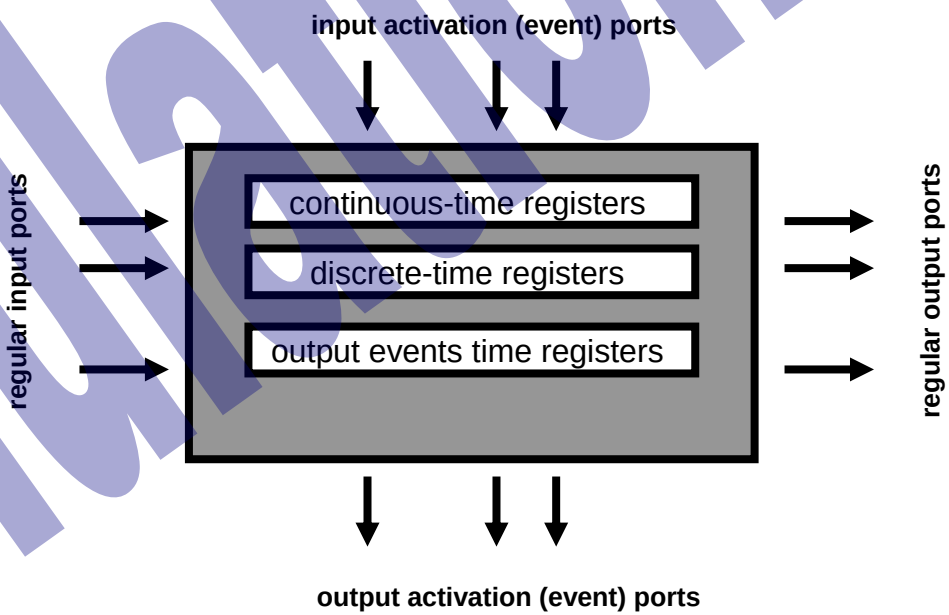
The final table

Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
Simulink/Stateflow	Stateflow and Simulink switches	Stateflow output selecting state evolution	No limitations	Conditions on Stateflow inputs and threshold crossing detector	Not supported	Integrator's reset from Stateflow output
Modelica	Not explicitly defined	Events enabling equations	No limitations	Triggering relations on variables (`when` statement)	Not an explicit language feature	Through `reinit` statement
HyVisual	Explicit finite state machine representation	Discrete-state refinement	No restrictions	Triggering conditions on state variables	Not supported	Assignment on the FSM edges

Scicos

- **Time set**: set of intervals in which a signal may change
  - Assumption: a signal **is constant outside** the time set





$\{x_1(t), T_1\} \quad \{x_2(t), T_2\}$

$$f(\ \{x_1(t), T_1\}, \{x_2(t), T_2\}\ ) = \{\ f(x_1(t), x_2(t)), (T_1 \cup T_2)\ \}$$

Simulation

The final table

Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
Simulink/Stateflow	Stateflow and Simulink switches	Stateflow output selecting state evolution	No limitations	Conditions on Stateflow inputs and threshold crossing detector	Not supported	Integrator's reset from Stateflow output
Modelica	Not explicitly defined	Events enabling equations	No limitations	Triggering relations on variables (`when` statement)	Not an explicit language feature	Through `reinit` statement
HyVisual	Explicit finite state machine representation	Discrete-state refinement	No restrictions	Triggering conditions on state variables	Not supported	Assignment on the FSM edges
Scicos	Not explicitly defined	Events switching dynamics	No restrictions	Threshold detectors	Threshold detectors	Reinitialization of integrators' state

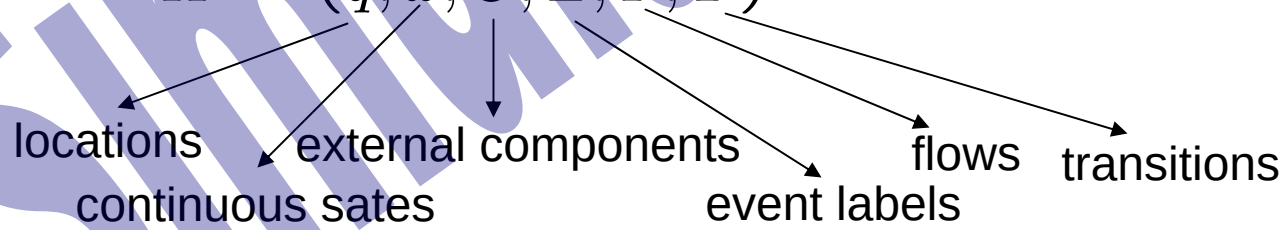
Shift

- **A world in shift is a set of hybrid components** $W = \{h_1, \cdots, h_w\}$
- **The world configuration is a tuple of the configurations of all components**

$$C_W = (C_{h_1}, \cdots, C_{h_w})$$

- **A component is a tuple**

$$H = (q, x, C, L, F, T)$$



Discrete phase: all enabled transitions are taken

Continuous phase: components evolve according to flows until a transition is enabled

Simulation

The final table

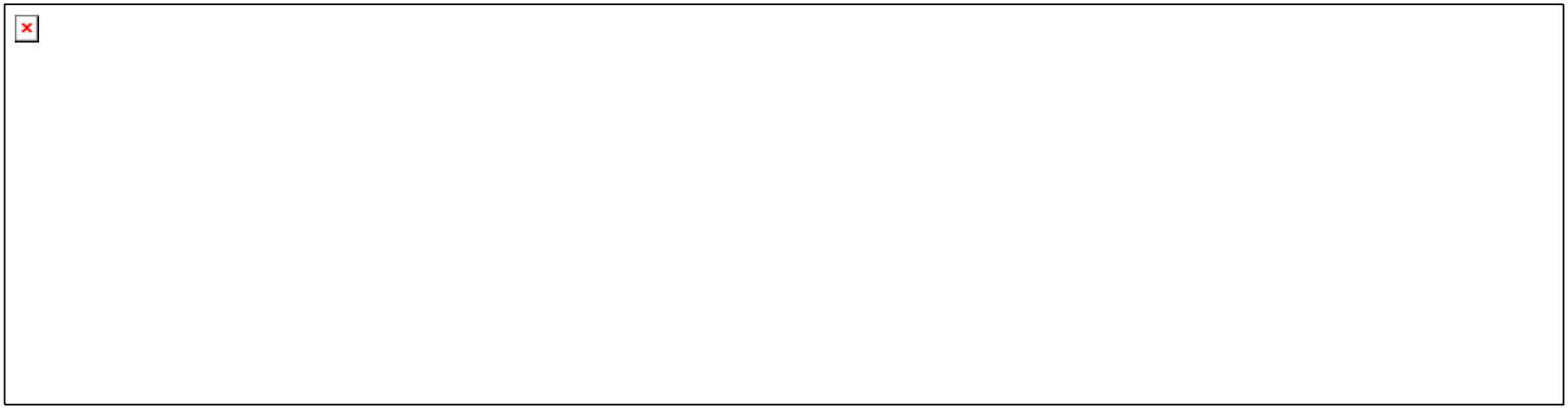
Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
Simulink/Stateflow	Stateflow and Simulink switches	Stateflow output selecting state evolution	No limitations	Conditions on Stateflow inputs and threshold crossing detector	Not supported	Integrator's reset from Stateflow output
Modelica	Not explicitly defined	Events enabling equations	No limitations	Triggering relations on variables (`when` statement)	Not an explicit language feature	Through `reinit` statement
HyVisual	Explicit finite state machine representation	Discrete-state refinement	No restrictions	Triggering conditions on state variables	Not supported	Assignment on the FSM edges
Scicos	Not explicitly defined	Events switching dynamics	No restrictions	Threshold detectors	Threshold detectors	Reinitialization of integrators' state
Shift	Textual definition of locations and transitions	Flows as locations' arguments	No Restrictions	Conditions on system variables	Conditions on system variables	Assignment statements

Charon

- Architectural hierarchy: Agents
- Behavioral hierarchy: Modes
  - Differential constraints
  - Algebraic constraints
  - Invariant constraints
  - Transitions
  - Entry and exit points
- Compositionality results

Simulation!

The final table



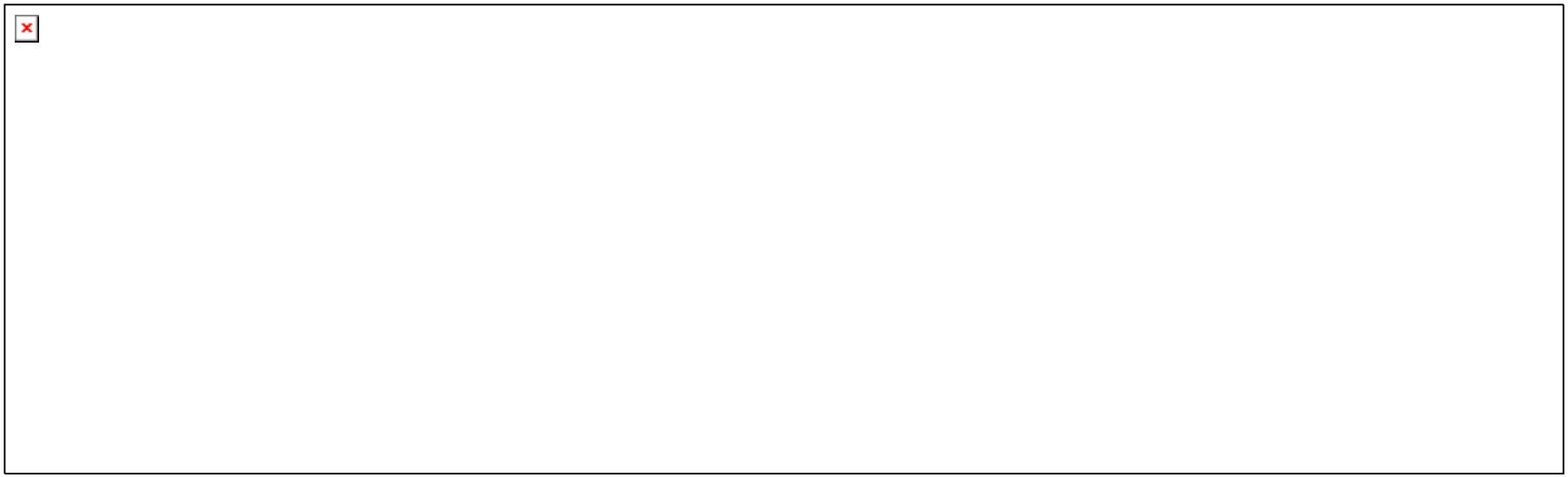
Hytech

- Linear Hybrid Automata
  - Invariant and guards are convex predicates over state variables X
  - Flows are convex predicates on $\dot{X}$
- Communication through shared variables and synchronization labels

- Reachability and Safety verification
- Parametric analysis

Verification

The final table



Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
HyTech	Explicit declaration of locations and transitions	Flows defined in each location	Convex predicate over the derivative of state variables	Conjunction of linear constraints	Convex predicate over state variables	Assignment statements

CheckMate

- **Switched Continuous System Block**

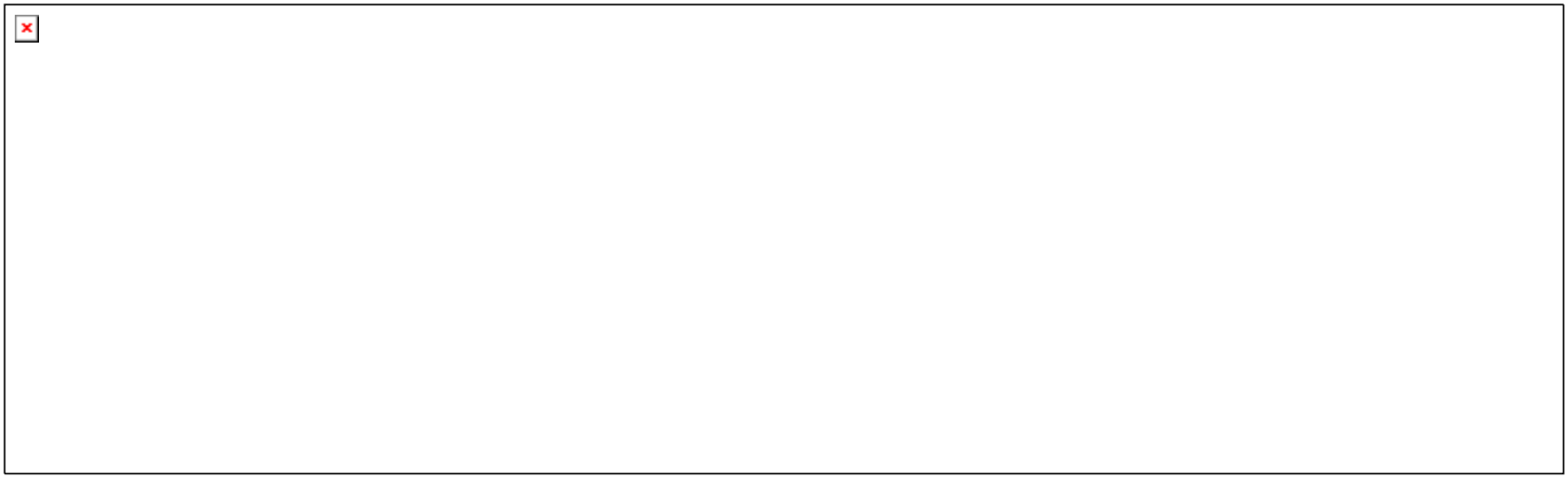
  $\dot{x} = f(x, \sigma)$, σ discrete vector

- **Polyhedral Threshold Block** $Cx \leq d$
- **Finite State Machine Block**

- **Reachability analysis through conservative approximation (QTS)**

Verification

The final table



Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
HyTech	Explicit declaration of locations and transitions	Flows defined in each location	Convex predicate over the derivative of state variables	Conjunction of linear constraints	Convex predicate over state variables	Assignment statements
CheckMate	Stateflow	Mode selector from Stateflow to a set of dynamics	Linear or non-linear (simulation only or approximation to linear dynamics)	Affine inequalities	Not supported	Affine maps

d/dt

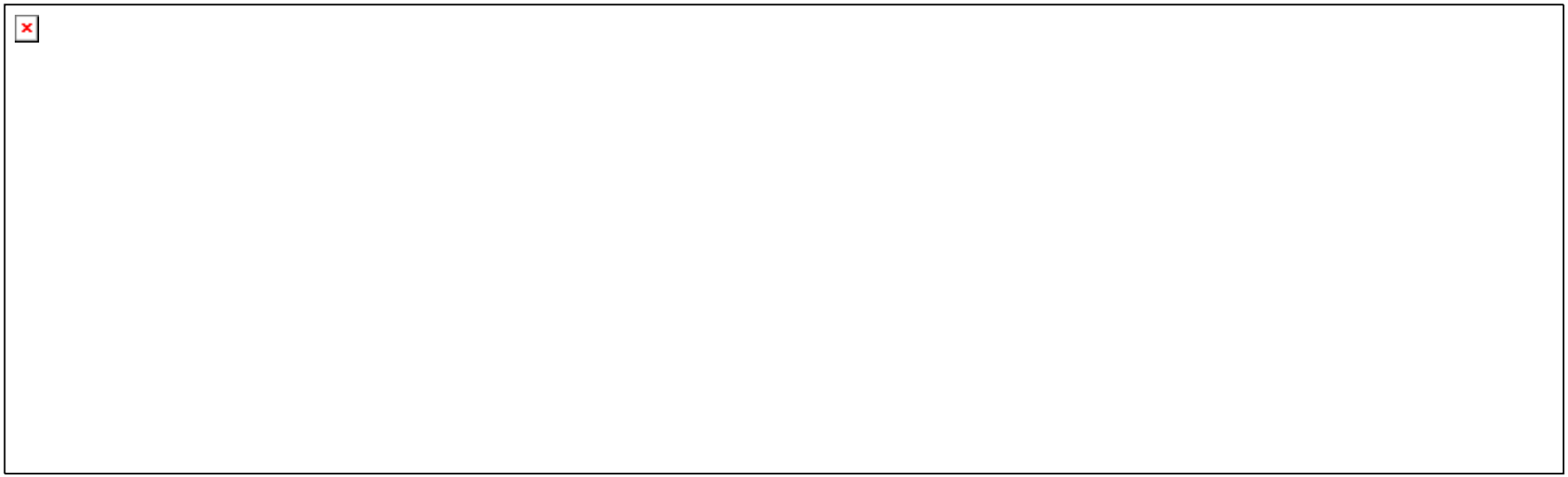
- **Continuous dynamics are linear**

$$\dot{x} = Ax + u$$

- **Invariants and guards are specified as convex polyhedra**

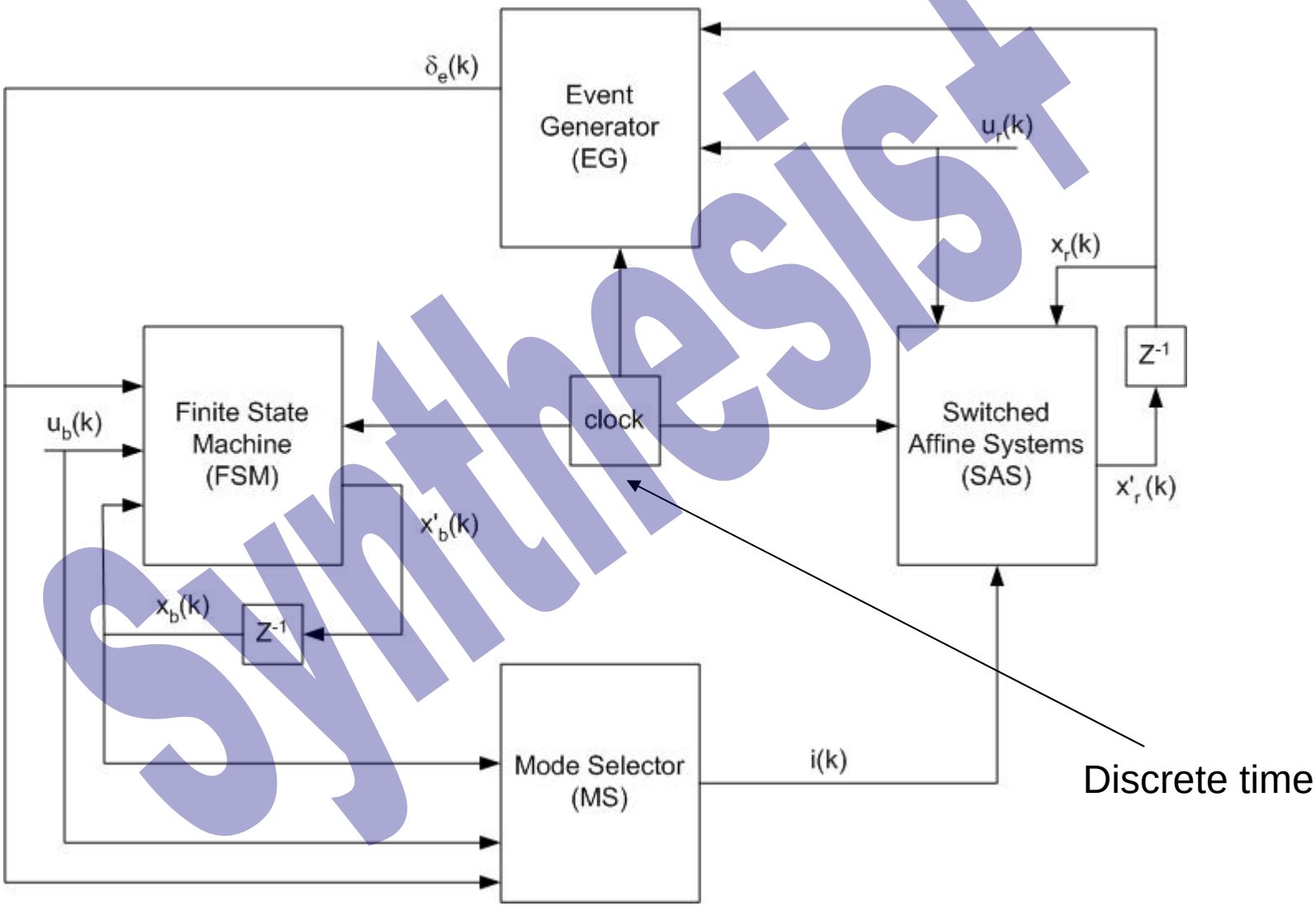
- **Reachability and Safety verification**

The final table



Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
HyTech	Explicit declaration of locations and transitions	Flows defined in each location	Convex predicate over the derivative of state variables	Conjunction of linear constraints	Convex predicate over state variables	Assignment statements
CheckMate	Stateflow	Mode selector from Stateflow to a set of dynamics	Linear or non-linear (simulation only or approximation to linear dynamics)	Affine inequalities	Not supported	Affine maps
d/dt	Explicit declaration of locations and transitions	Flows defined in each location	Linear	Convex polyhedra	Convex polyhedra	Not supported in the version shipped to us

Hysdel



Synthesis+

Hysdel Synthesis Results

- **PWA plant**

$$\begin{aligned} x(k+1) &= A_i x(k) + B_i u(k) + f_i \\ s.t. &\quad L_i x(k) + E_i u(k) \leq W_i \\ if &\quad [x'(k)u'(k)]' \in \mathcal{D}_i, \qquad i \in I \end{aligned}$$

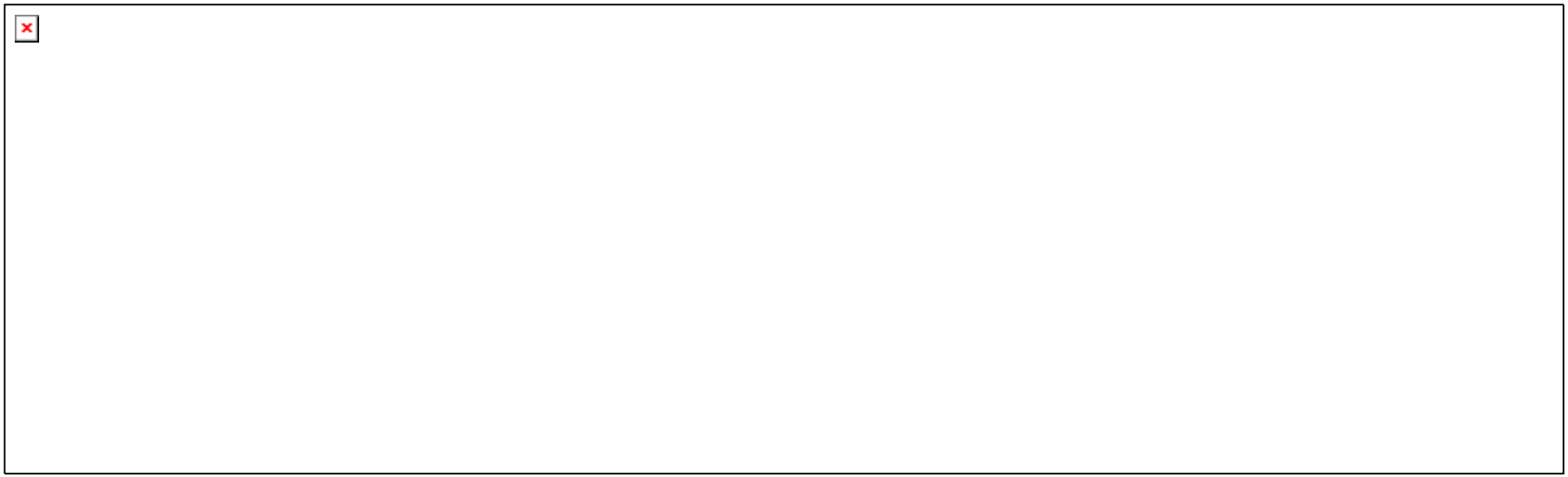
- **Cost**

$$\begin{aligned} J_N^*(x(0)) &= \min_{u_0,\ldots,u_{N-1}} ||Q_f x(N)||_l + \sum_{k=0}^{N-1} ||Ru(k)||_l + ||Qx(k)||_l \\ subj. \quad to \quad & x(k+1) = f_{PWA}(x(k), u(k)) \\ & x(N) \in \mathcal{X}_{set} \\ & if\ l = 2,\ then\ Q = Q' \succeq 0, \quad Q_f = Q_f' \succeq 0, \quad R = R' \succeq 0 \end{aligned}$$

Theorem 1 *Consider the constrained finite-time optimal control problem. Then, the set of feasible parameters $\mathcal{X}_f^N$ is convex and the optimizer $U_N^* : \mathcal{X}_f^N \rightarrow \mathbb{R}^{Nm}$ is continuous and piecewise affine, i.e.*

$$\begin{aligned} U_N^*(x(0)) = F_r x(0) + G_r \quad if \quad x(0) \in \mathcal{P}_r &= \{x \in \mathbb{R}^n | H_r x \leq K_r\}, \\ r &= 1, \ldots, R \end{aligned}$$

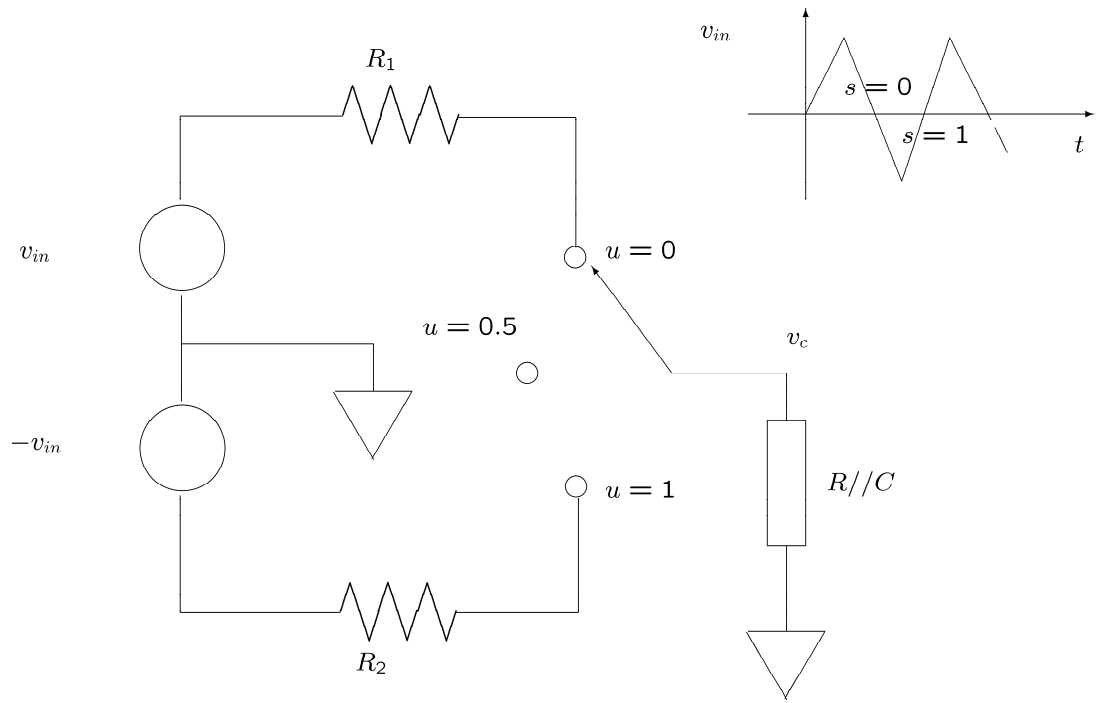
The final table



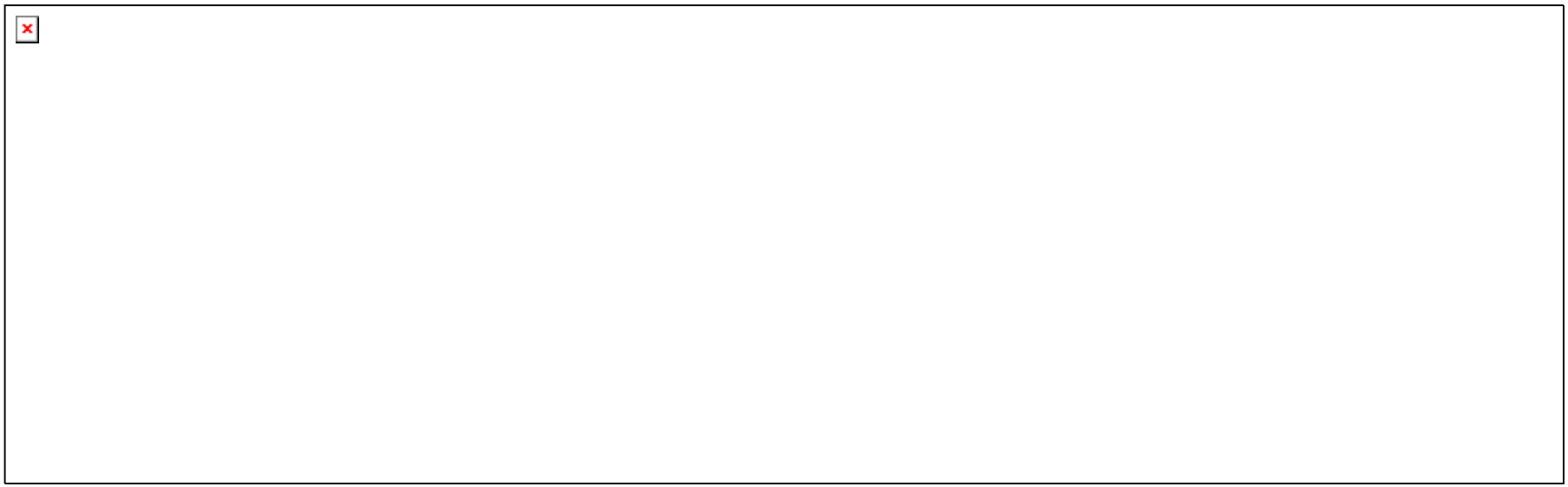
Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
HyTech	Explicit declaration of locations and transitions	Flows defined in each location	Convex predicate over the derivative of state variables	Conjunction of linear constraints	Convex predicate over state variables	Assignment statements
CheckMate	Stateflow	Mode selector from Stateflow to a set of dynamics	Linear or non-linear (simulation only or approximation to linear dynamics)	Affine inequalities	Not supported	Affine maps
d/dt	Explicit declaration of locations and transitions	Flows defined in each location	Linear	Convex polyhedra	Convex polyhedra	Not supported in the version shipped to us

Name	Automata Definition	State-to-Dynamics Mapping	Supported Dynamics	Guards	Invariants	Reset Maps
Hysdel	Logic formulas on Boolean variables	Mode selectors	Discrete Time and Linear	Threshold conditions on system variables	Not supported	Modeled as one step dynamics

Synthesis Example



Other features



Name	Hierarchy	Composition	OO	Causality	Algebraic Loops	Continuous/Discrete Interface
HyTech	No	Synchronization of automata and shared variables	No	Causal	Not supported	Locations associated with flows and reset maps
CheckMate	No	No	No	Causal	Not supported	Mode selectors switching dynamics and affine reset maps
d/dt	No	No	No	Causal	Not supported	Location associated with flows
Hysdel	No	No	No	Causal	Not supported	Mode selectors switching dynamics