Hybrid Systems
Modeling and Analysis

Radu Grosu
Vienna University of Technology

Acknowledgements

Thanks for providing slides or inspiration:

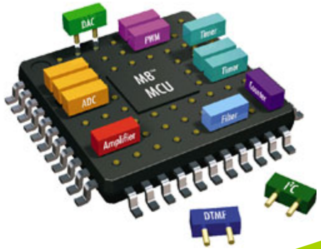
Rajeev Alur, Calin Belta, Antoine Girard, Tom Henzinger
Oded Maler, George Pappas, Ulrich Schmid, Paulo Tabuada

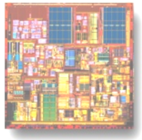
Thanks for helping preparing the slides:

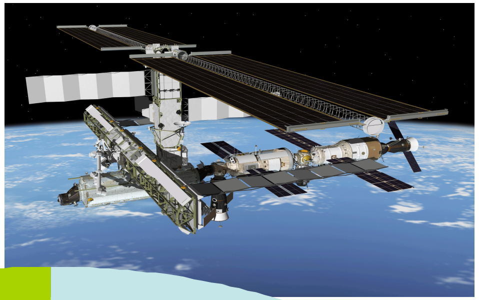
Ezio Bartocci, Richard de Francisco
Ariful Islam, Abhishek Murthy

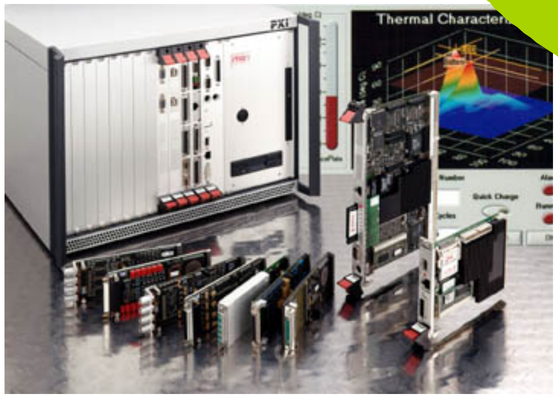
A word of caution:

Only partial coverage and reference
Biased towards my research interests









Hybrid Systems

Systems on a chip

Networked embedded systems

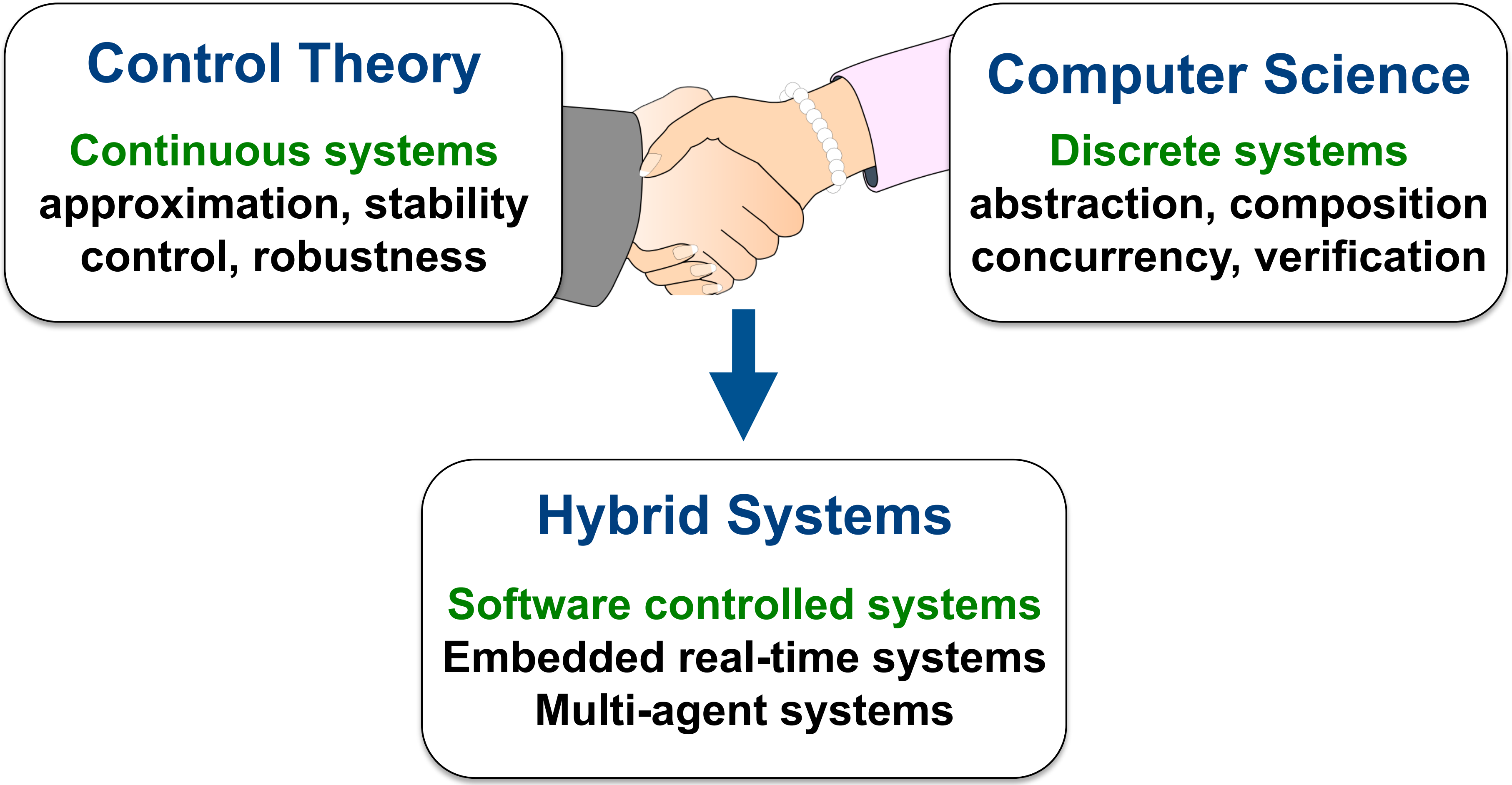
Real-Time systems

Fault-tolerant systems

System architectures

HW/SW codesign

Continuous AND Discrete Systems



Control Theory

Continuous systems

approximation, stability
control, robustness

Computer Science

Discrete systems

abstraction, composition
concurrency, verification

Hybrid Systems

Software controlled systems

Embedded real-time systems
Multi-agent systems

Models and Tools

Dynamic systems with continuous & discrete state variables

	Continuous Part	Discrete Part
Models	Differential equations, transfer functions,	Automata, Petri nets, Statecharts,
Analytical Tools	Lyapunov functions, eigenvalue analysis,	Boolean algebra, formal logics, verification,
Software Tools	Matlab, Matrix_x, VisSim,	Statemate, Rational Rose, SMV,

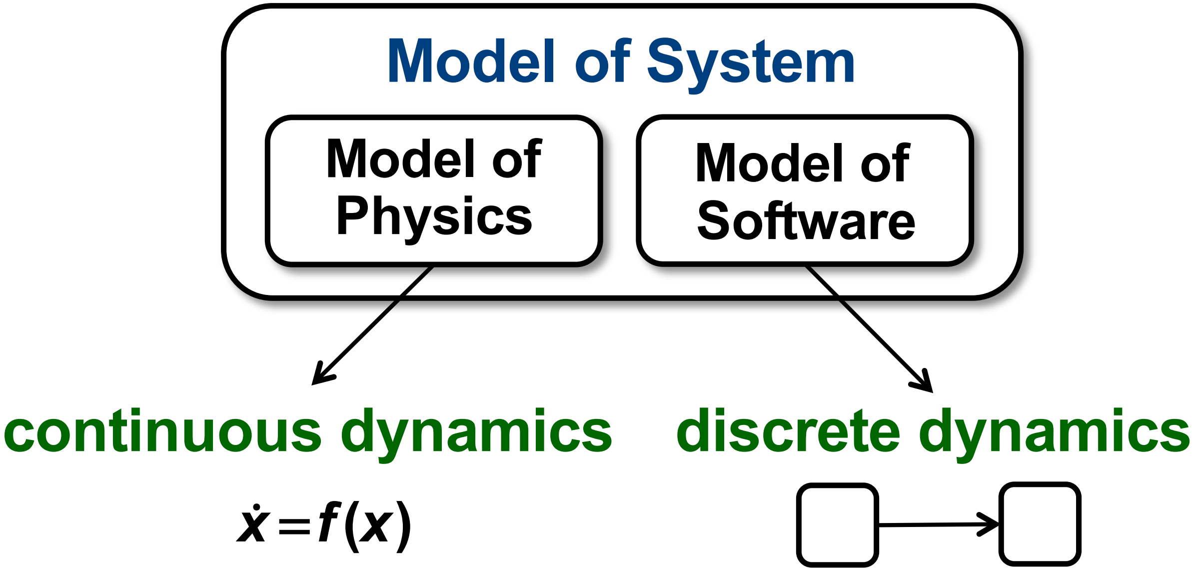
Outline

Modeling of hybrid systems

Analysis of hybrid systems

Appplication to cardiac dynamics

Modeling a Hybrid System



Model of System

Model of Physics

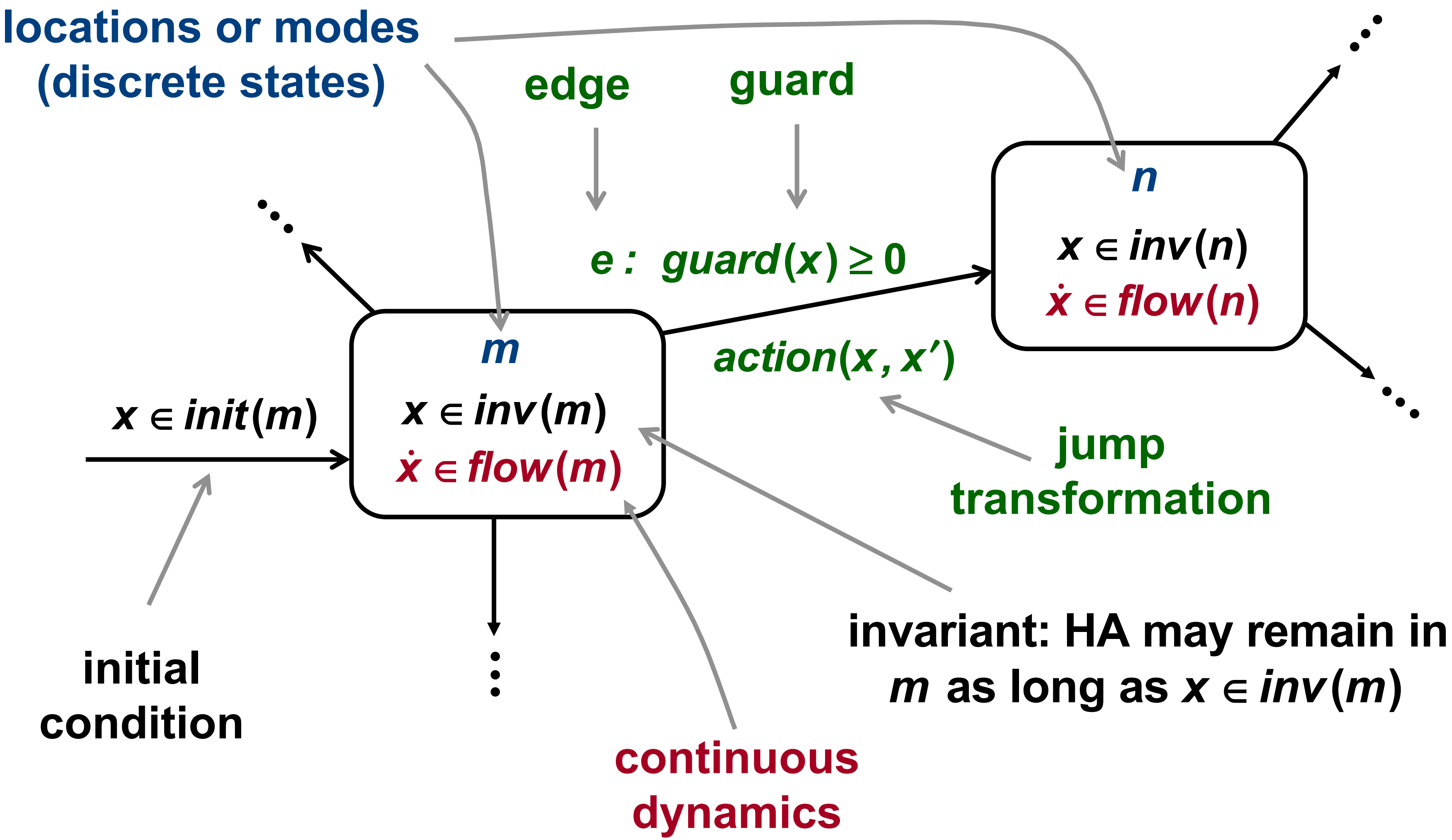
Model of Software

continuous dynamics

$$\dot{x} = f(x)$$

discrete dynamics

Hybrid Automaton (HA)



locations or modes (discrete states)

edge

guard

$e: \ guard(x) \geq 0$

$action(x, x')$

m

$x \in inv(m)$

$\dot{x} \in flow(m)$

n

$x \in inv(n)$

$\dot{x} \in flow(n)$

$x \in init(m)$

jump transformation

invariant: HA may remain in m as long as $x \in inv(m)$

initial condition

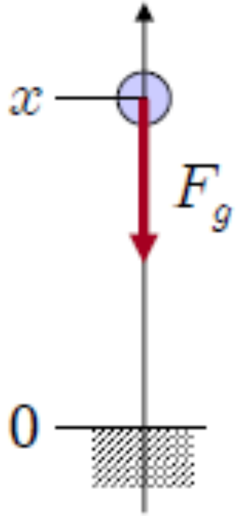
continuous dynamics

Example: Bouncing Ball

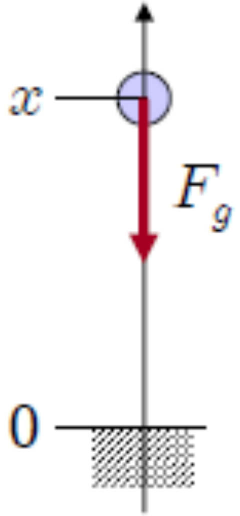
Ball has mass m and position x

Ball initially at position x_0 and at rest

Ball bounces when hitting ground at $x = 0$



Bouncing Ball: Free Fall



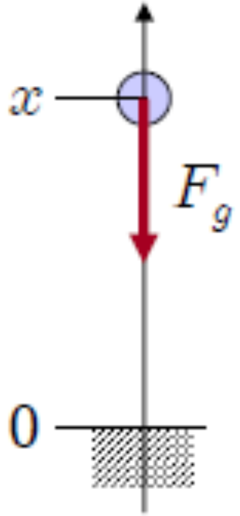
Condition for free fall: $x \geq 0$

Physical law: $F_g = -mg = -m\ddot{x}$

Differential equations: $\dot{x} = v$

First order $\dot{v} = -g$

Bouncing Ball: Bouncing

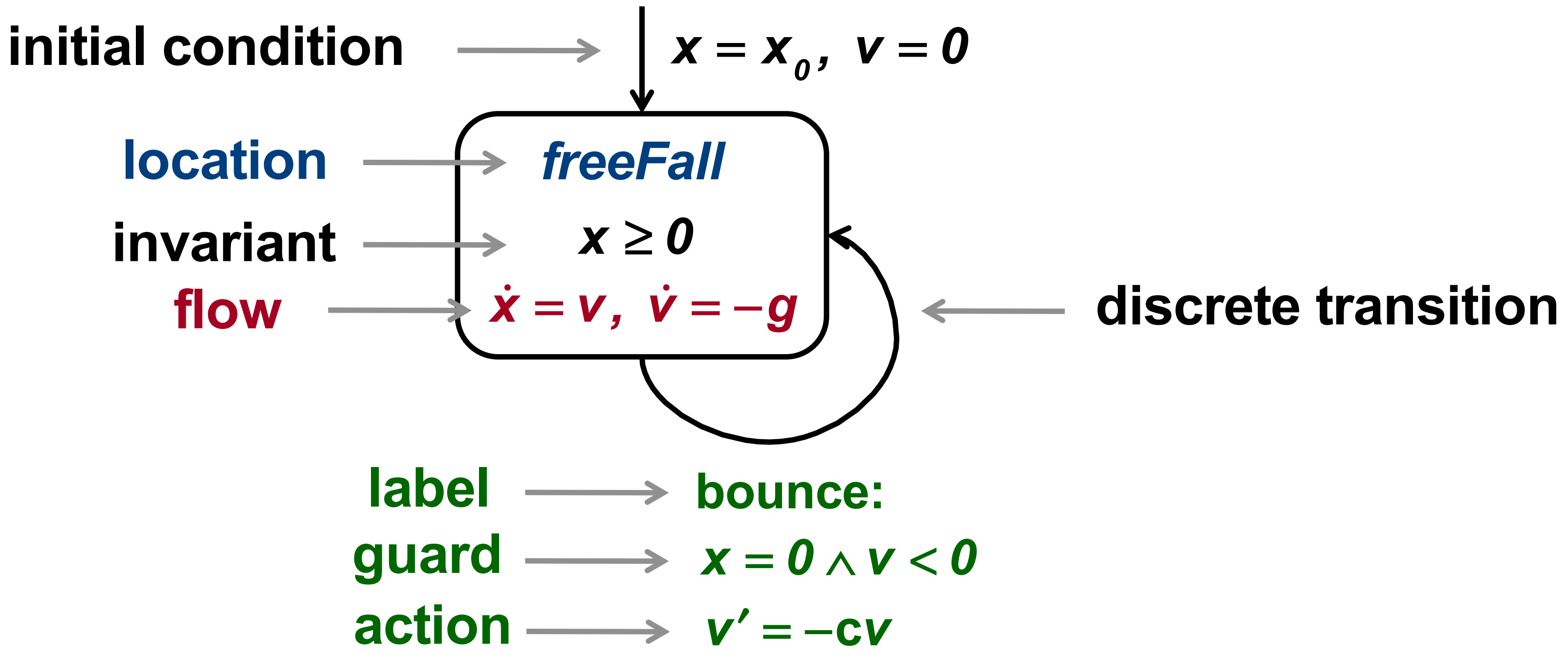


Condition for bouncing: $x = 0$

Action for bouncing: $v' = -cv$

Coefficient c: deformation, friction

Bouncing Ball: Hybrid Automaton



initial condition → $x = x_0,\ v = 0$

location → *freeFall*

invariant → $x \geq 0$

flow → $\dot{x} = v,\ \dot{v} = -g$

discrete transition

label → bounce:

guard → $x = 0 \wedge v < 0$

action → $v' = -cv$

Hybrid Automaton $\mathcal{H}$

Variables: Continuous variables $x = [x_1, \ldots, x_n]$

Control Graph: Finite directed multigraph (V,E)

- **Finite set** V of control modes
- **Finite set E** lof control switches

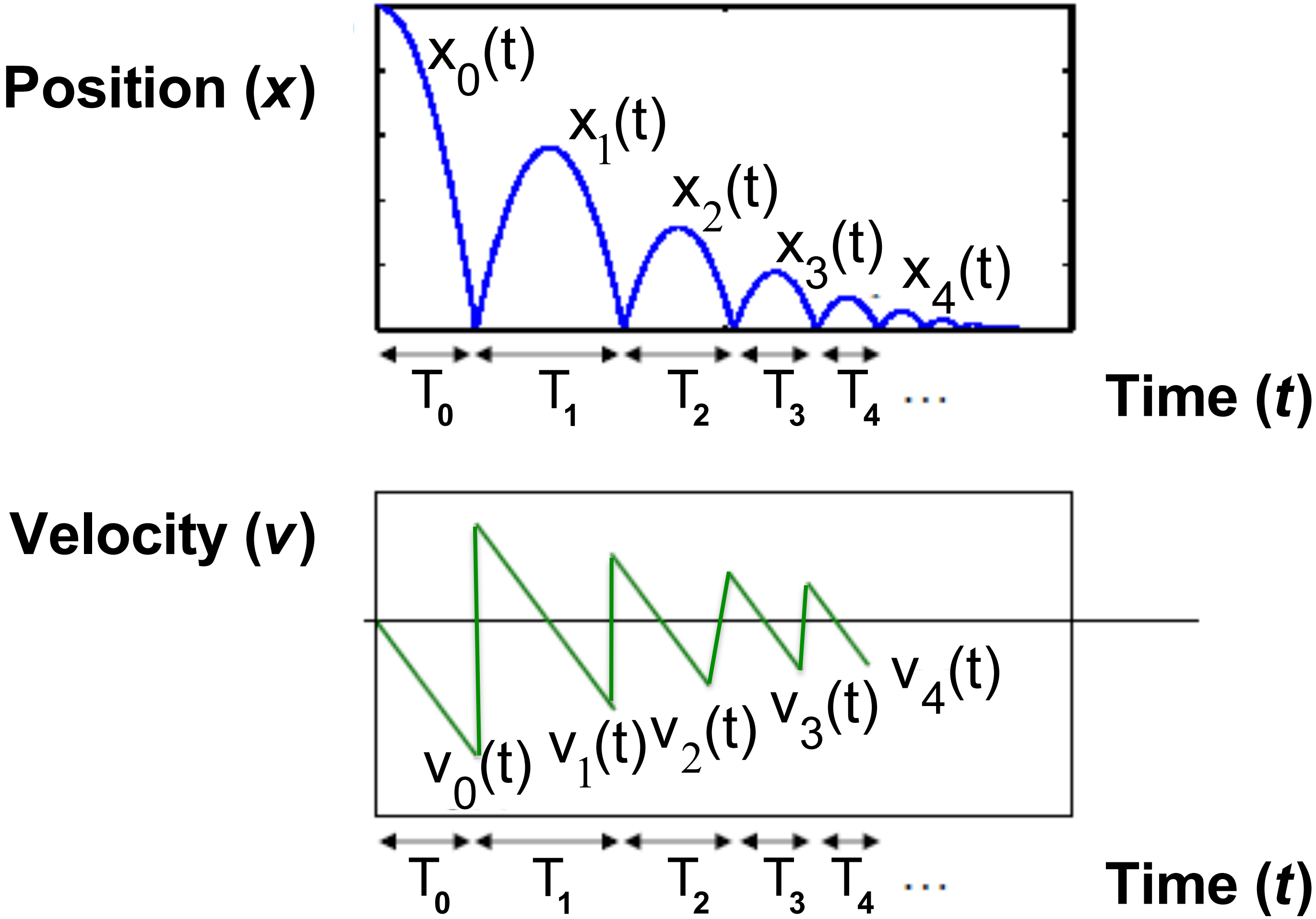
Vertex labeling functions: for each $v \in V$

- **Initial states:** init(v)(x) defines initial region
- **Invariant:** inv(v)(x) defines invariant region
- **Continuous dynamics:** $\dot{x}$ is in flow(v)(x)

Edge labeling functions: for each $e \in E$

- **Guard:** guard(e)(x) defines enabling region
- **Update:** action(e)(x, x') defines the reset region
- **Synchronization labels:** label(e) defines communication

Execution of Bouncing Ball



Executions of a Hybrid Automaton

State: (m,x) such that $x \in \mathrm{inv}(m)$

Initialization: (m,x) such that $x \in \mathrm{init}(m)$

Two types of state updates:

Discrete switches: $(m,x) \rightarrow^{a} (m',x')$ if $\exists\ e \in E.$

$$\mathrm{label}(e) = a \ \wedge\ \mathrm{guard}(e)(x) \geq 0$$

$$\mathrm{action}(x,x')$$

Continuous flows: $(m,x) \rightarrow^{f} (m',x')$ if $\exists\ v \in V.\ f : [0,\mathrm{T}] \rightarrow \mathbb{R}.$

$$f(0) = x \ \wedge\ f(\mathrm{T}) = x'$$

$$\forall 0 \leq t \leq \mathrm{T}.\ \ f(t) \in \mathrm{inv}(v) \ \wedge\ \dot{f}(t) \in \mathrm{flow}(v)(f(t))$$

Outline

Modeling of hybrid systems

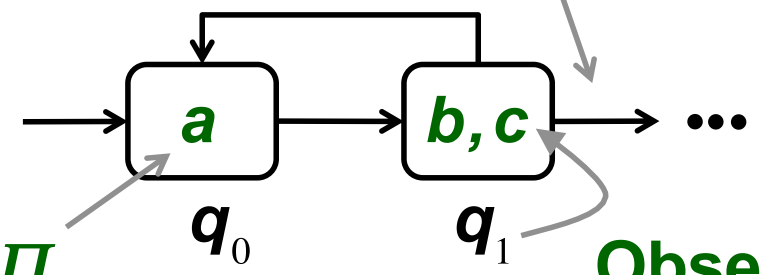
Analysis of hybrid systems

Appplication to cardiac dynamics

Infinite Transition Structures

Transition function

$\delta : Q \rightarrow (\wp(Q) - \varnothing)$



Observables Π

Finite set

Observation function

$\ll . \gg \; : Q \rightarrow \wp(\Pi)$

States Q

Infinite set

∞ Transition Structure $\mathcal{K}_H = (Q, V, \ll . \gg, \delta)$

States Q: Pairs (m,x) consisting of

Discrete modes: $m \in V$

Continuous variables: $x \in \mathbb{R}^n$

Observables V: $\ll m,x \gg \; = \; m$

Transition relation δ: $(m',x') \in \delta(m,x)$ iff

Discrete switches: $\exists e \in E.\ a \in L.$

$e = (m, m') \;\wedge\; \text{label}(e) = a$

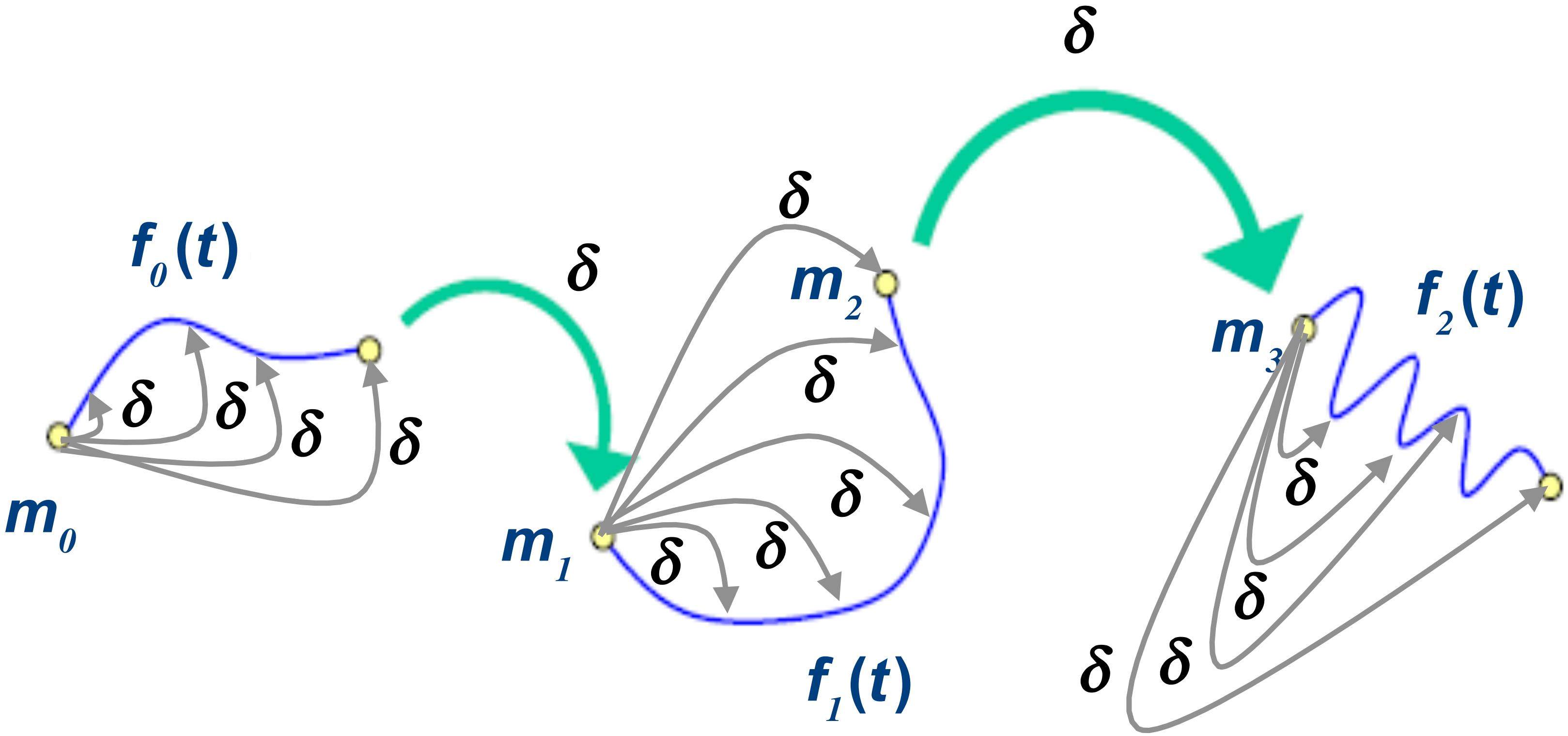
$\text{guard}(e)(x) \geq 0 \;\wedge\; \text{action}(x,x')$

Continuous flow: $\exists\, v \in V.\ f : [0,\mathrm{T}] \rightarrow \mathbb{R}.$

$f(0) = x \;\wedge\; f(\mathrm{T}) = x'$

$\forall 0 \leq t \leq \mathrm{T}.\ \ f(t) \in \text{inv}(v) \;\wedge\; \dot{f}(t) \in \text{flow}(v)(f(t))$

Intuition of the Construction

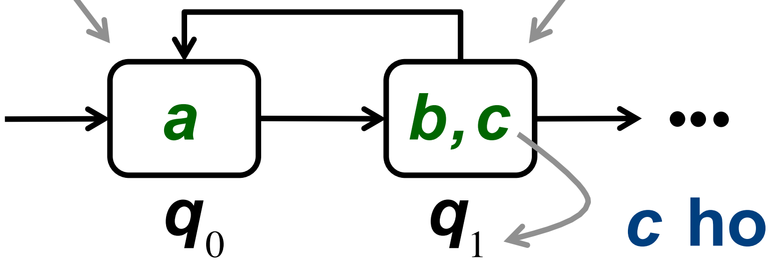


Time is abstracted away

Semantics of Transition Structures

Predecessor of q_1
$q_0 \in \delta^{-1}(q_1)$

Successor of q_0
$q_1 \in \delta(q_0)$



c holds at q_1
$c \in \ll q_1 \gg$

q_0-run of $\mathcal{K}$: $r = q_0 q_1 \dots \quad q_{i+1} \in \delta(q_i)$

Trace of r: $\ll r \gg \; = \; \ll q_0 \gg \ll q_1 \gg \dots \; = \; \{a\}\{b,c\}\dots$

L^q: **All q-runs,** $\ll L^q \gg$: **The associated traces**

Trends in Software Design

Emerging notations: UML-RT, Statecharts/Statfelow

- Visual
- Hierarchical modeling of control flow

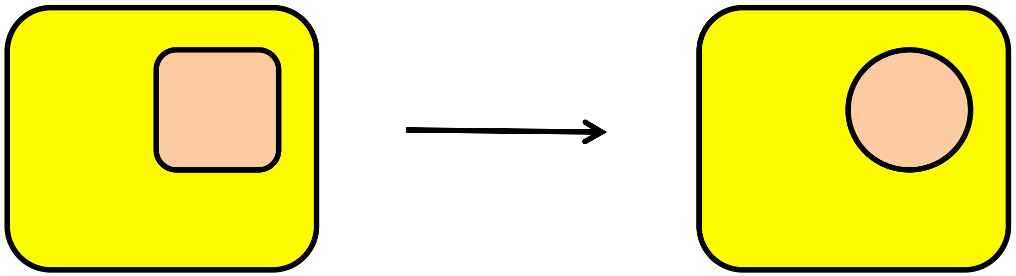
Prototyping/modeling but no analysis

- Ad-hoc, informal features
- No support for abstraction

CHARON modeling and simulation language

- Formal, hierarchical
- Hybrid-automata based

Composable Behavioral Interfaces



Which properties are preserved?

Can we restrict reasoning to modified parts of design?

- Components should have precise interface specification
- Components differing only in internal details are equivalent

Observational Semantics

Classical PL concept of denotational semantics

- Two programs are equivalent if they compute the same function

For reactive systems one has to

- Account for the ongoing interaction (behavior over time)

Observational semantics of a hybrid component

- **Static interface:** Set of input/output variables
- **Behavioral interface:** Set of traces

Compositional Semantics

Traces should retain all (but not more) information needed to

- Determine the interaction of a component with other components

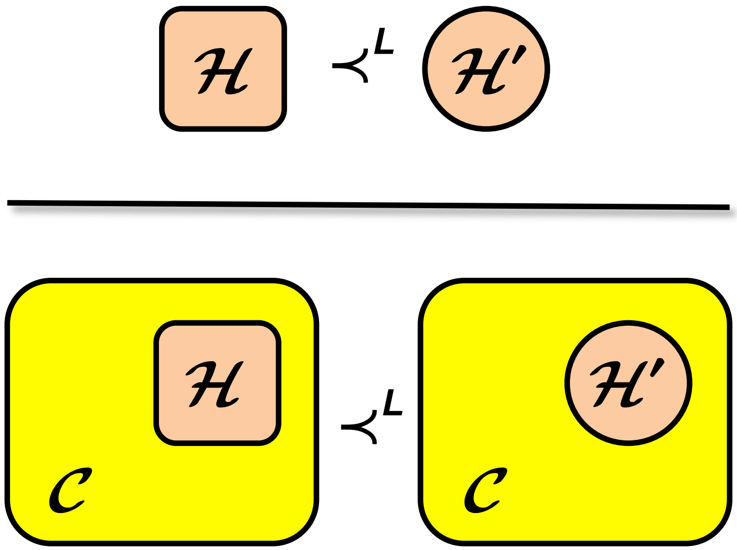
Desired theorems

- Substitutivity: $\mathcal{H} \prec^L \mathcal{H}' \;\Rightarrow\; \forall$context C. $C[\mathcal{H}] \prec^L C[\mathcal{H}']$
- Compositionality: $[\![\mathcal{H}||\mathcal{H}']\!] = [\![\mathcal{H}]\!] \cap [\![\mathcal{H}']\!]$

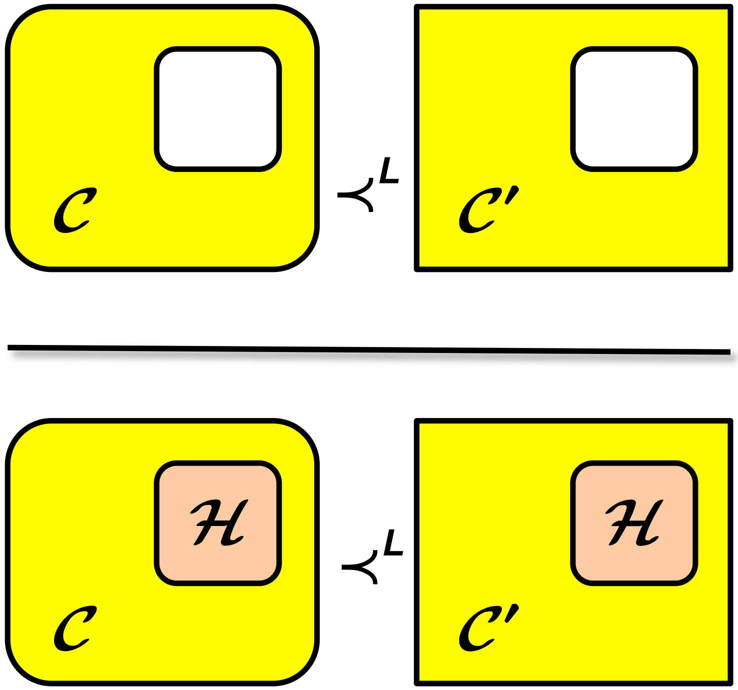
Typically one can

- Project out information about private variables and modes
- But not timing information and flows of communication variables

Compositional Reasoning



Sub-mode refinement



Context refinement

Quotient Structures

Atomic State-Equivalence: $p \cong^A q$ iff $\ll p \gg \; = \; \ll q \gg$

Atomic Regions A: The equivalence classes of $\cong^A$

Quotient $\mathcal{K}_{/\cong} = (Q_{/\cong}, \Pi, \ll . \gg_{/\cong}, \delta_{/\cong})$: If $\cong$ refines $\cong^A$

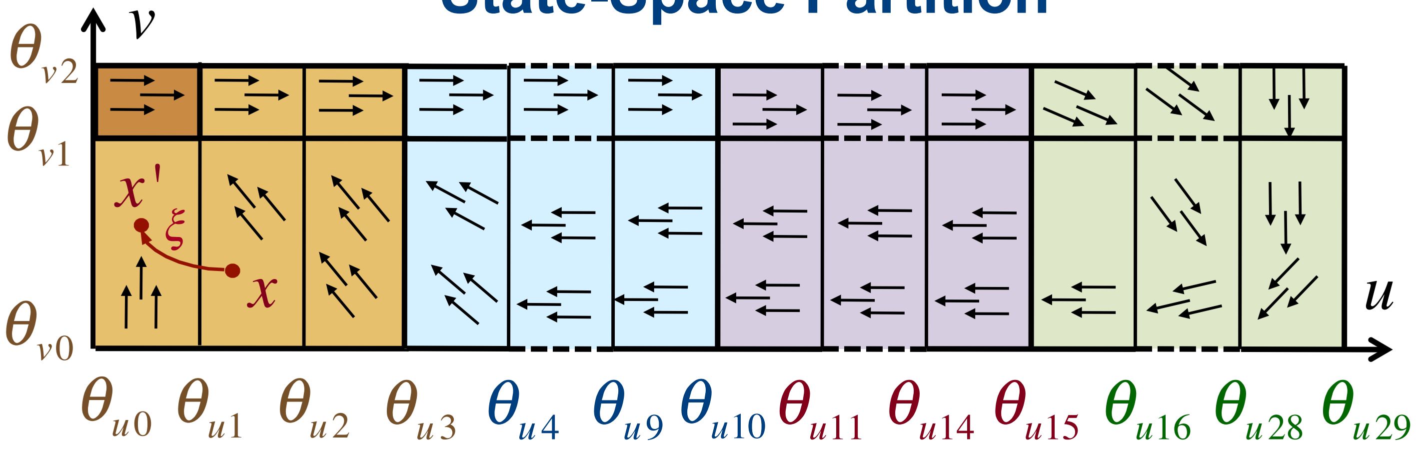
$Q_{/\cong}$ is the set of equivalence classes of $\cong$

$R \in \delta_{/\cong}(P)$ if $\exists p \in P . r \in R.\; r \in \delta(p)$

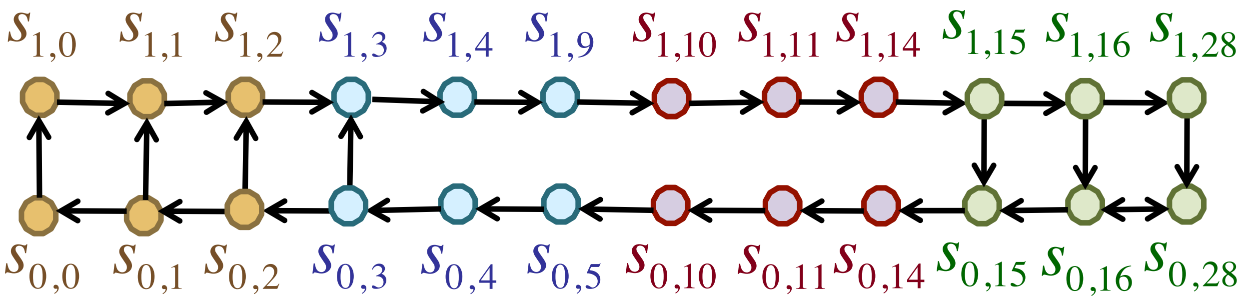
$\pi \in \; \ll R \gg_{/\cong}$ if $\exists r \in R.\; \pi \in \; \ll r \gg$

Example of Quotient Structures

State-Space Partition



Finite Quotient Structure



State Logic

State logic $\mathcal{L}$: Formulas interpreted over states of $\mathcal{K}$

$[\![\varphi]\!]_{\mathcal{K}}$: Set of states of $\mathcal{K}$ satisfying φ

$\mathcal{L}$ - MC problem: For $\varphi \in \mathcal{L} \wedge q \in Q_{\mathcal{K}}$ is $q \in [\![\varphi]\!]_{\mathcal{K}}$?

Fully Abstract Semantics

$\mathcal{L}$ - equivalence: $p \cong^{\mathcal{L}} q$ **if** $\forall \varphi \in \mathcal{L}.\ \ p \in [\![\varphi]\!]_{\mathcal{K}}$ **iff** $q \in [\![\varphi]\!]_{\mathcal{K}}$

$\mathcal{L}$ admits abstraction: $\forall \cong$ **ref** $\cong^{\mathcal{L}} .\forall \varphi \in \mathcal{L}.\ [\![\varphi]\!]_{\mathcal{K}} = \bigcup [\![\varphi]\!]_{\mathcal{K}/\cong}$

$\cong^{\mathcal{L}}$ **is the fully abstract semantics for** $\mathcal{L}$

$\cong$ **has finite index** $\Rightarrow$ **finite-state MC**

Trace Equivalence

Trace containtment: A binary relation on Q

$p \prec_l q$ implies $\ll L^p \gg \; \subseteq \; \ll L^q \gg$

Any trace containtment: $p \prec^L q$ implies $\exists \prec_l \, . \; p \prec_l q$

Trace equivalence: $p \cong^L q$ implies $p \prec^L q \; \wedge \; q \prec^L p$

Thm: $\mathcal{K}$ satisfies an LTL formula iff $\mathcal{K}_{/\cong^L}$ satisfies it

Bisimulation

Simulation: $p \prec_s q$ implies

(bc) $\ll p \gg \; = \; \ll q \gg$

(is) $\forall p' \in \delta(p).\ \exists q' \in \delta(q).\ p' \prec_s q'$

Bisimulation $\cong^b$ **:** A symmetric simulation

Bisimilarity $\cong^B$ **:** $p \cong^B q$ if $\exists \cong^b .\ p \cong^b q$

Thm: $\mathcal{K}$ satisfies an CTL formula iff $\mathcal{K}_{/\cong^B}$ satisfies it

Symbolic Theory

Region: A possibly infinite set of states of $\mathcal{K}$

Symbolic theory $\mathcal{T}_{\mathcal{K}} = (\Sigma, [\![.]\!])$ of $\mathcal{K}$: A tuple

Region representatives Σ: finite representation of regions in $\mathcal{K}$

Extension function $[\![.]\!] : \Sigma \to \wp(Q)$: Maps representatives to regions

(atm) $\forall R \in A.\ \exists \sigma_R \in \Sigma.\ [\![\sigma_R]\!] = R$ (Σ_A)

(pre) $\forall \sigma \in \Sigma.\ \exists \mathrm{Pre}(\sigma) \in \Sigma.\ [\![\mathrm{Pre}(\sigma)]\!] = \left\{ q \in Q \mid \delta(q) \cap [\![\sigma]\!] \neq \varnothing \right\}$

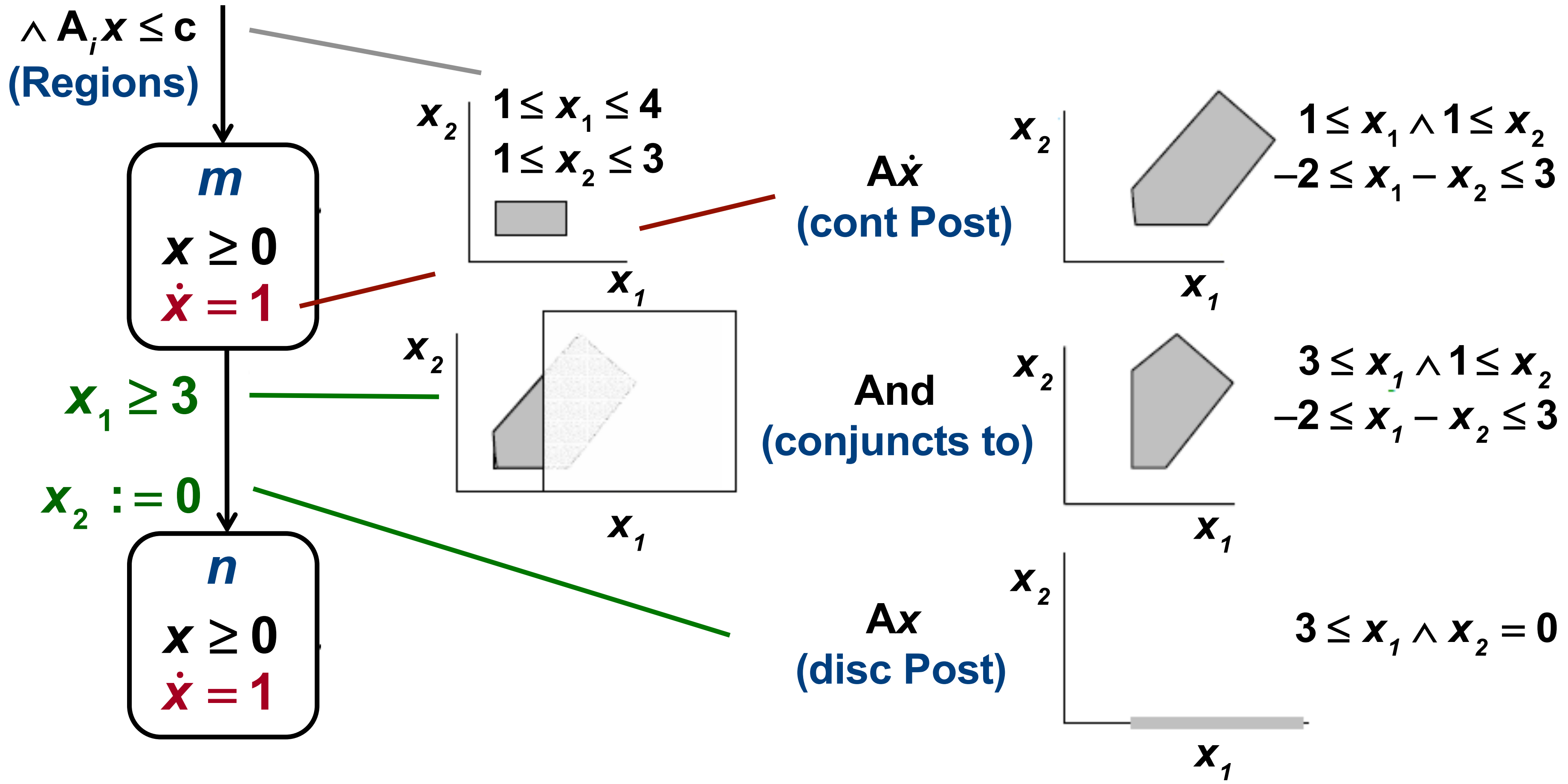
(and) $\forall \sigma, \tau \in \Sigma.\ \exists \mathrm{And}(\sigma, \tau) \in \Sigma.\ [\![\mathrm{And}(\sigma, \tau)]\!] = [\![\sigma]\!] \cap [\![\tau]\!]$

(diff) $\forall \sigma, \tau \in \Sigma.\ \exists \mathrm{Diff}(\sigma, \tau) \in \Sigma.\ [\![\mathrm{Diff}(\sigma, \tau)]\!] = [\![\sigma]\!] - [\![\tau]\!]$

(∅?) $\exists$computable Empty: $\Sigma \to \mathbb{B}$. Empty(σ) iff $[\![\sigma]\!] = \varnothing$

(in?) $\exists$computable In: $Q \times \Sigma \to \mathbb{B}$. In(q,$\sigma$) iff $q \in [\![\sigma]\!]$

Symbolic Theory: Polyhedral Sets



$\wedge A_i x \le c$
(Regions)

m
$x \ge 0$
$\dot{x} = 1$

$x_1 \ge 3$

$x_2 := 0$

n
$x \ge 0$
$\dot{x} = 1$

$1 \le x_1 \le 4$
$1 \le x_2 \le 3$

$A\dot{x}$
(cont Post)

$1 \le x_1 \wedge 1 \le x_2$
$-2 \le x_1 - x_2 \le 3$

And
(conjuncts to)

$3 \le x_1 \wedge 1 \le x_2$
$-2 \le x_1 - x_2 \le 3$

Ax
(disc Post)

$3 \le x_1 \wedge x_2 = 0$

Thus $(m, 1 \le x_1 \le 4 \wedge 1 \le x_2 \le 3) \rightarrow^{\delta} (n, 3 \le x_1 \wedge x_2 = 0)$

Symbolic Semi-Algorithms

SSA: **Input:** $\mathcal{T}_{\mathcal{K}}$, **Output:** **region representatives**

Backward reachability $\mathcal{A}_{\Diamond}$**:**

Input: $\sigma_0 \in \Sigma_A$

Induction: $\sigma_{i+1} = \mathrm{Pre}(\sigma_i)$

Termination: $\bigcup_{i \leq i \leq k+1} [\![\sigma_i]\!] \subseteq \bigcup_{i \leq i \leq k+1} [\![\sigma_i]\!]$

Symbolic Semi-Algorithms

SSA: **Input:** $\mathcal{T}_{\mathcal{K}}$, **Output:** region representatives

Partition refinement $\mathcal{A}_{PR}$:

- **Input:** $\mathcal{S}_0 = \Sigma_A$
- **Induction:** $\mathcal{S}_{i+1} = \mathcal{S}_i \cup \{\mathrm{Pre}(\sigma), \mathrm{And}(\sigma,\tau), \mathrm{Diff}(\sigma,\tau) \mid \sigma,\tau \in \mathcal{S}_i\}$
- **Termination:** $\{[\![\sigma]\!] \mid \sigma \in \mathcal{S}_{i+1}\} \subseteq \{[\![\sigma]\!] \mid \sigma \in \mathcal{S}_i\}$

Quotient automaton for CTL-MC

Symbolic Semi-Algorithms

SSA: **Input:** $\mathcal{T}_{\mathcal{K}}$, **Output:** region representatives

Observation refinement $\mathcal{A}_{OR}$:

Input: $\mathcal{S}_0 = \Sigma_A$

Induction: $\mathcal{S}_{i+1} = \mathcal{S}_i \cup \{\text{Pre}(\sigma), \text{And}(\sigma, \tau) \mid \sigma \in \mathcal{S}_i \wedge \sigma \in \Sigma_A\}$

Termination: $\{[\![\sigma]\!] \mid \sigma \in \mathcal{S}_{i+1}\} \subseteq \{[\![\sigma]\!] \mid \sigma \in \mathcal{S}_i\}$

Quotient automaton for LTL-MC

Infinite-State MC

Step 1: Get a finite bisimilar/trace-equivalence quotient

Step 2: Apply a standard CTL/LTL model checker

Combine MC with partition/observation refinement

μ-Calculus

Formulas of μ-calculus:

$$\varphi ::= \pi \mid \neg\pi \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \wedge \varphi_2 \mid \exists\bigcirc\varphi \mid \forall\bigcirc\varphi \mid \mu X.\varphi \mid \upsilon X.\varphi$$

π is an observable

X is a propositional variable

μ is the least-fixpoint operator

υ is the gratest-fixpoint operator

$$\exists\Diamond\pi ::= \mu X.(\pi \vee \exists\bigcirc X)$$

μ-Calculus MC-Algorithm

```
procedure $\mu$SSA($\mathcal{T}_{\mathcal{K}}$,$\varphi$) {
    input: $\mathcal{T}_{\mathcal{K}}$, $\varphi$;   output: $\sigma \in \Sigma$ such that $[\![\sigma]\!] = [\![\varphi]\!]$
    case $\varphi \in \Sigma_A$: return $\varphi$
    case $\varphi = \varphi_1 \wedge \varphi_2$: return And($\mu$SSA($\mathcal{T}_{\mathcal{K}}$,$\varphi_1$), $\mu$SSA($\mathcal{T}_{\mathcal{K}}$,$\varphi_1$))
    case $\varphi = \neg\varphi_1$: return Diff(true, $\mu$SSA($\mathcal{T}_{\mathcal{K}}$,$\varphi_1$))
    case $\varphi = \exists\bigcirc\varphi_1$: return Pre($\mu$SSA($\mathcal{T}_{\mathcal{K}}$,$\varphi_1$))
    case $\varphi = \mu X.\varphi_1$:  $\sigma_1$ = false; repeat
        $\sigma_2$ := $\sigma_1$; $\sigma_1$ = $\mu$SSA($\mathcal{T}_{\mathcal{K}}$,$\varphi_1$[X := $\sigma_2$]);
        until $[\![\sigma_1]\!] = [\![\sigma_2]\!]$; return $\sigma_1$
}
```

Timed Automata

Continuous variables: timers $\dot{x} = 1$

Invariants and guards: $x < c$, $x \geq c$

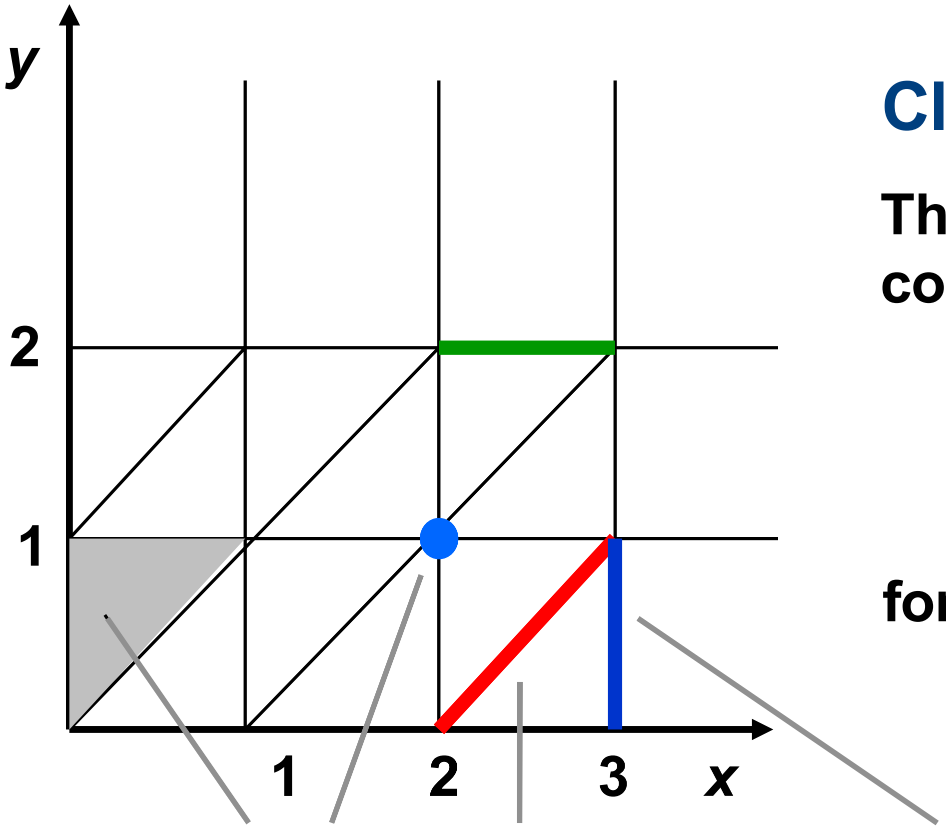
Actions: $x := 0$

Can express: lower and upper bounds on delays

Thm: $\mathcal{K}_{/\cong^{B}}$ has finite index.

Crl: μCalculus MCP is decidable (and effective).

Finite Bisimulation Quotient



An equivalence class (region)

Clock equivalence $w \cong w'$

They satisfy the same set of constraints of the form:

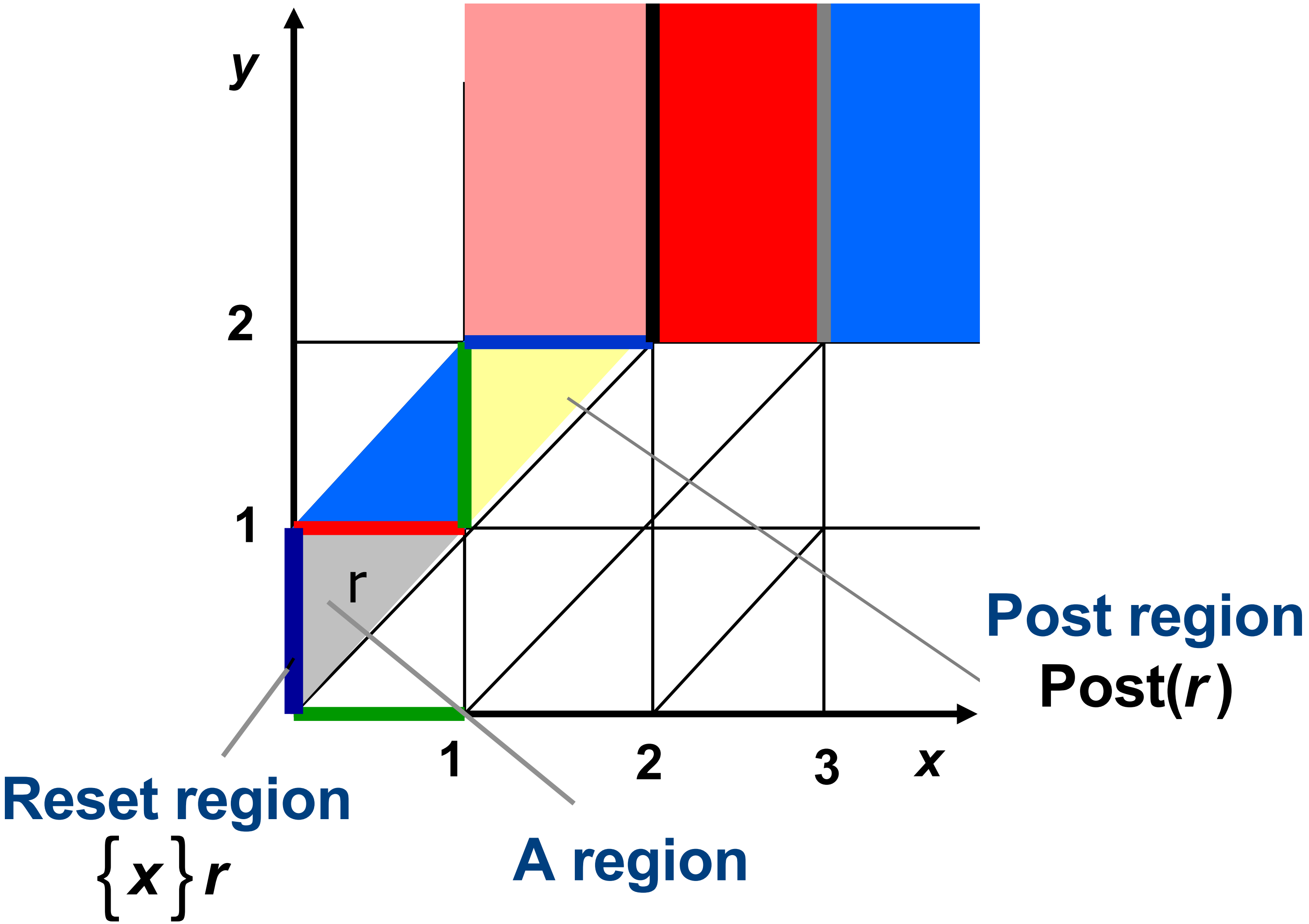
$$x_i < c, \ \ x_i = c$$

$$x_i - x_j < c, x_i - x_j = c$$

for c ≤ largest cst relevant to x_i

Regions ∝ (# locations) × (× all-constants) × (# clocks)!

Region Operations



Initialized Multi-Rate Automata

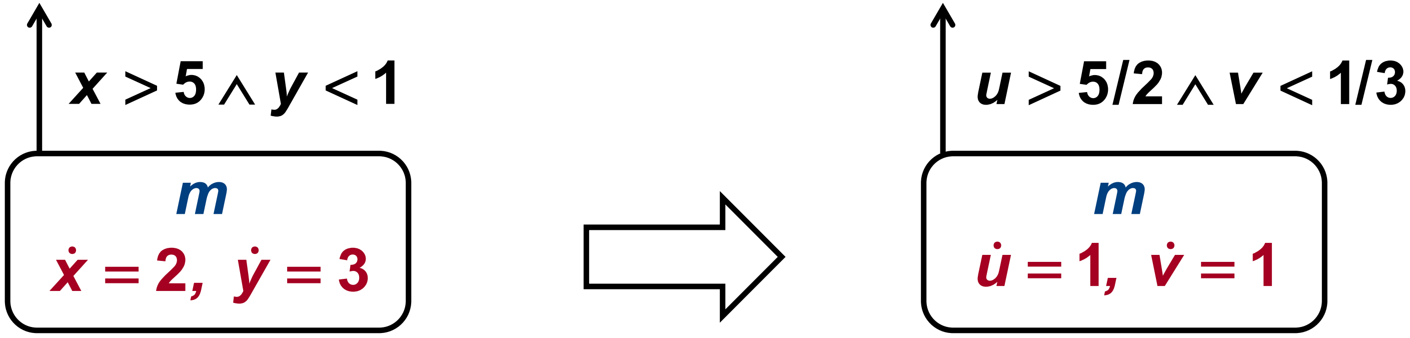
Continuous variables: $\dot{x} = c$

Invariants and guards: $x < c,\ x \geq c$

Actions: $x_i := c_j$ (when rate of x_i changes)

Thm: $\mathcal{K}_{/\cong^B}$ has finite index.

Crl: μCalculus MCP is decidable (and effective).



Sharp Decidability Bounds

Initialization is essential when changing rate!

Thm: The reachibility problem for not initialized
2-rate hybrid automata is undecidable

Prf: Encode 2-counter machines with 2-rate automata

Other extensions: undecidable as well

- Inequalities on variables : $x_i \leq x_j$
- Initialization with variables: action(e,x)$_i$ = x_j for $i \neq j$

Initialized Rectangular Automata

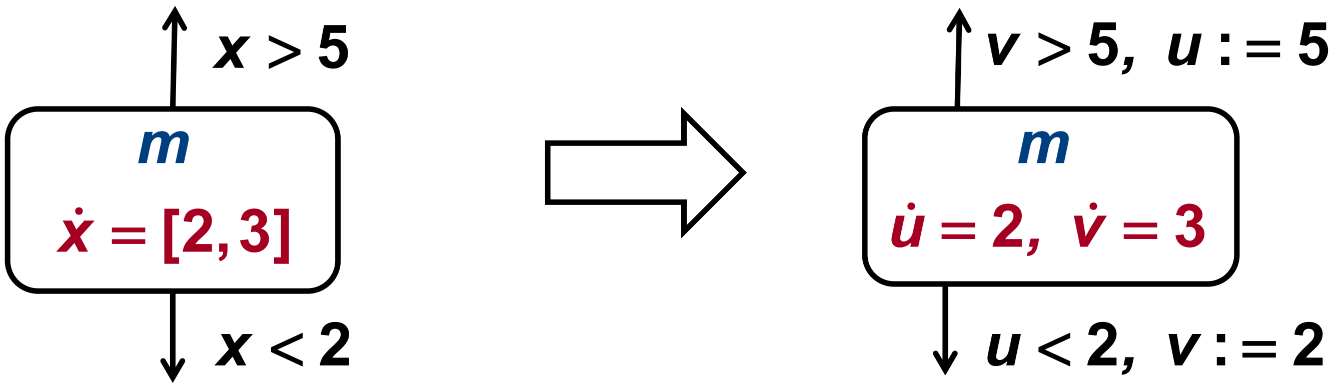
Continuous variables: $\dot{x} = [c,d]$

Invariants and guards: $x < c,\ x \geq c$

Actions: $x_i := c_j$ (when rate of x_i changes)

Thm: $\mathcal{K}_{/\cong^{L}}$ has finite index.

Crl: $\mathrm{LTL} - \mathrm{MCP}$ is decidable (and effective).



Continuous Systems Alone

Given: $\mathcal{K}$ has region $r_0 \subseteq \mathbb{R}^n$, $\dot{x} = f(x)$

Question: does $\mathcal{K}_{/\equiv^B}$ have finite index?

Spiral counter-example:

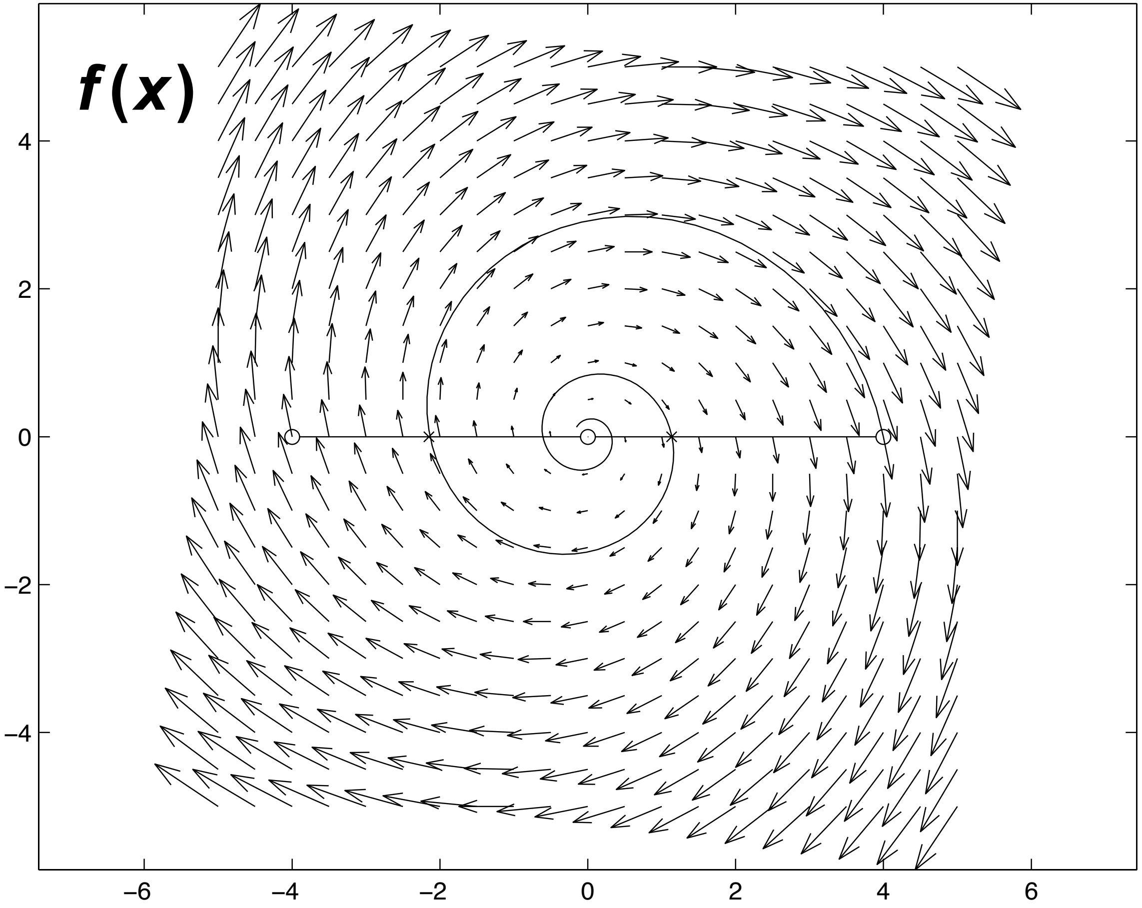
Initial region r_0 given by

$$-5 \leq x \leq 0 \wedge 0 \leq x \leq 5$$

Dynamics is given by

$$\dot{x} = Ax$$

where A has both real and imaginary parts in its eigenvalues (invar)



O-Minimal Structures

A structure over $\mathbb{R}$ is order-minimal if:

- $\forall$ definable set is a finite $\bigcup$ of points and open intervals

O-minimal structures:

- Polyhedral sets: $\mathbb{R}$ with $\leq, +, -, 0, 1$
- Semialgebraic sets and e^x: $\mathbb{R}$ with $\leq, +, -, \times, e^x, 0, 1$
- Many much more: such as sub-analytic

O-Minimal Hybrid Automata

Guards, flows, invariants: def in same oM-structure

Edges: reset all variables to constants or intervals

Thm: $\mathcal{K}_{/\cong^{B}}$ has finite index.

Crl: μCalculus MCP is decidable (and effective).

Linear Hybrid Automata

Continuous variables: $A\dot{x} \leq c$

Invariants and guards: $Ax \leq c$

Actions: $x := Ax$

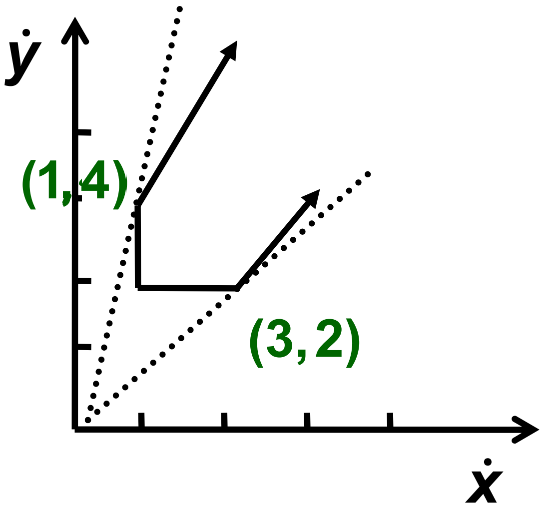
Symbolic representation: polyhedra

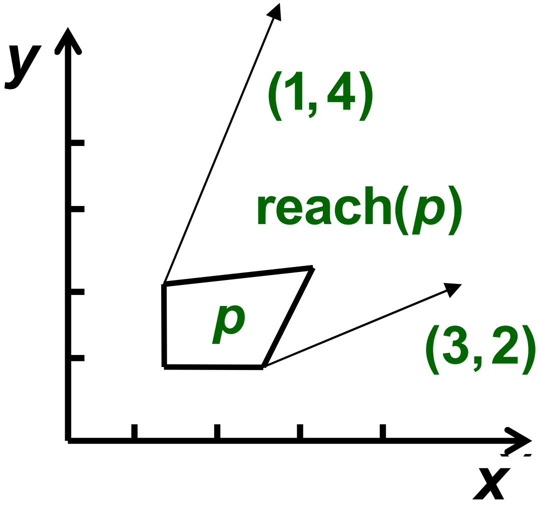
Reachability: A semi-algorithm

Methodolgy: Abstract dynamics by diff inclusions

Tools: HyTech, Phaver, SpaceEx

Computing Time Successor





Thm: If init, inv and flow are polyhedra then the set of reachable states is a computable polyhedron

Alg: Apply extremal rates to vertices of init

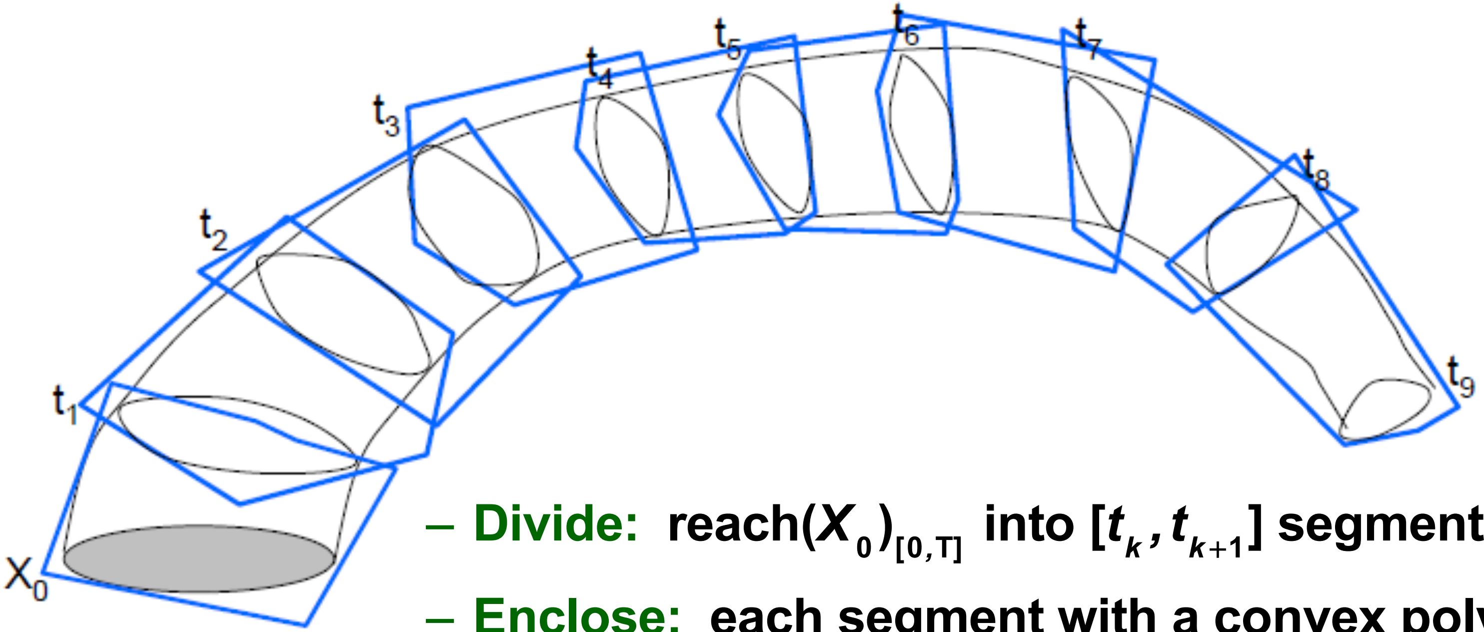
Approximating Reachability

Given:

- Initial region: Polyhedron X_0
- Flow equation: $\dot{x} = f(x)$

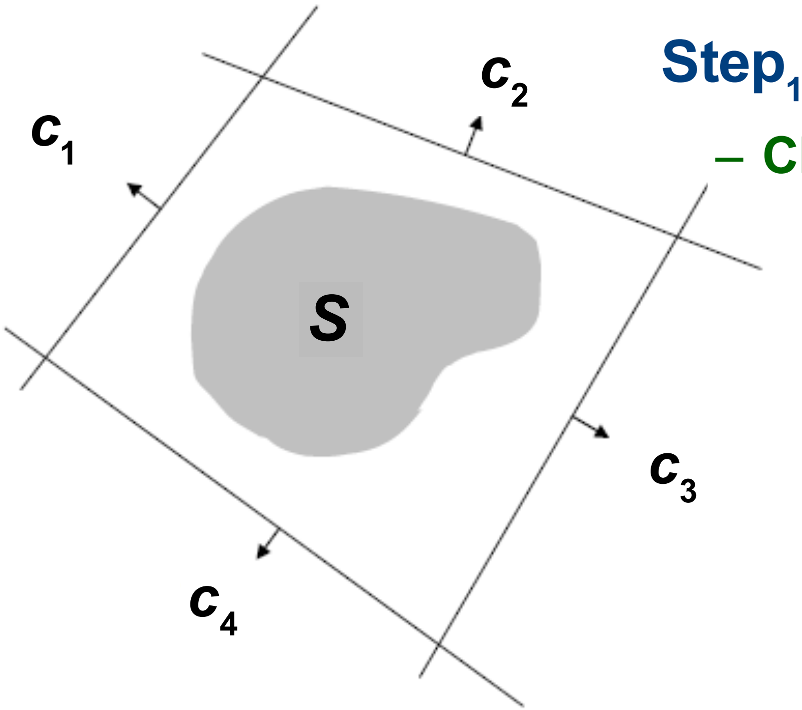
Conservatively approximate: reach(X_0)

Polyhedral Flow Approximation



- **Divide:** reach$(X_0)_{[0,T]}$ into $[t_k, t_{k+1}]$ segments
- **Enclose:** each segment with a convex polytope
- **Union:** reach$(X_0)_{[0,T]}^{App} =$ union of polytope

Wrapping Hyper-planes Around a Set



Step$_1$:

- **Choose:** normal vectors $c_1 \ldots c_n$

Wrapping a Flow Pipe Segment

Step$_2$: Wrap $\text{reach}_{[t_k, t_{k+1}]}(X_0)$ in a polytope by solving for each i the optimisation problem

$$d_i = \max_{x_0, t} c_i^T x(t, x_0)$$

$$x_0 \in X_0$$

$$t \in [t_k, t_{k+1}]$$

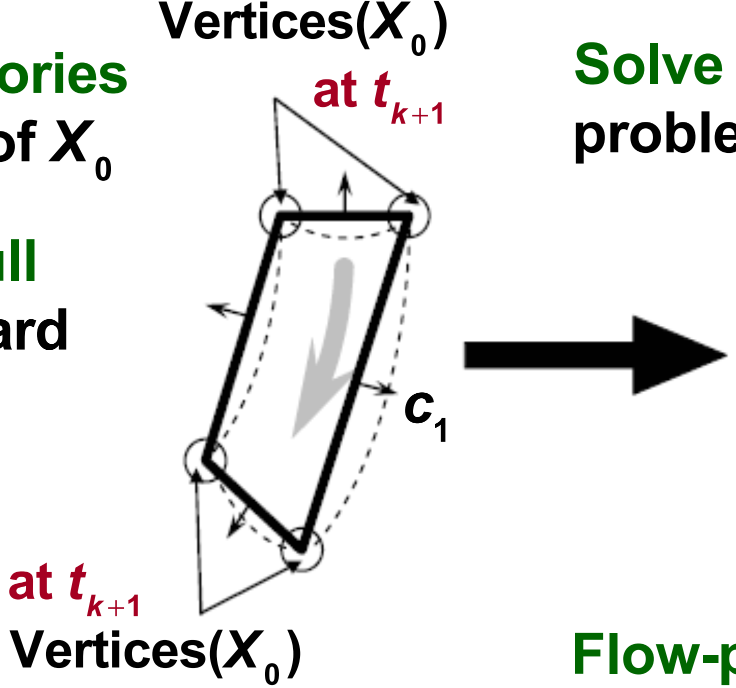
Optimization problem is solved by embedding simulation into objective function computation

Flow Pipe Segment Approximation

Step 1:

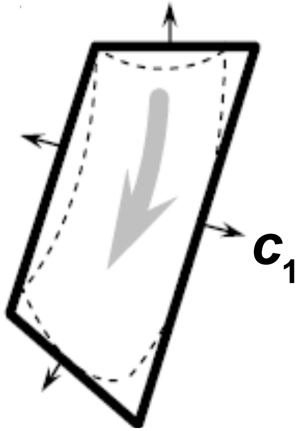
a. Simulate trajectories from each vertex of X_0

b. Take convex hull and identify outward normal vectors c_i



Step 2:

Solve optimization problem for each d_i



Flow-pipe segment approximated by

$$\left\{ x \mid c_i^T x \leq d_i, \forall i \right\}$$

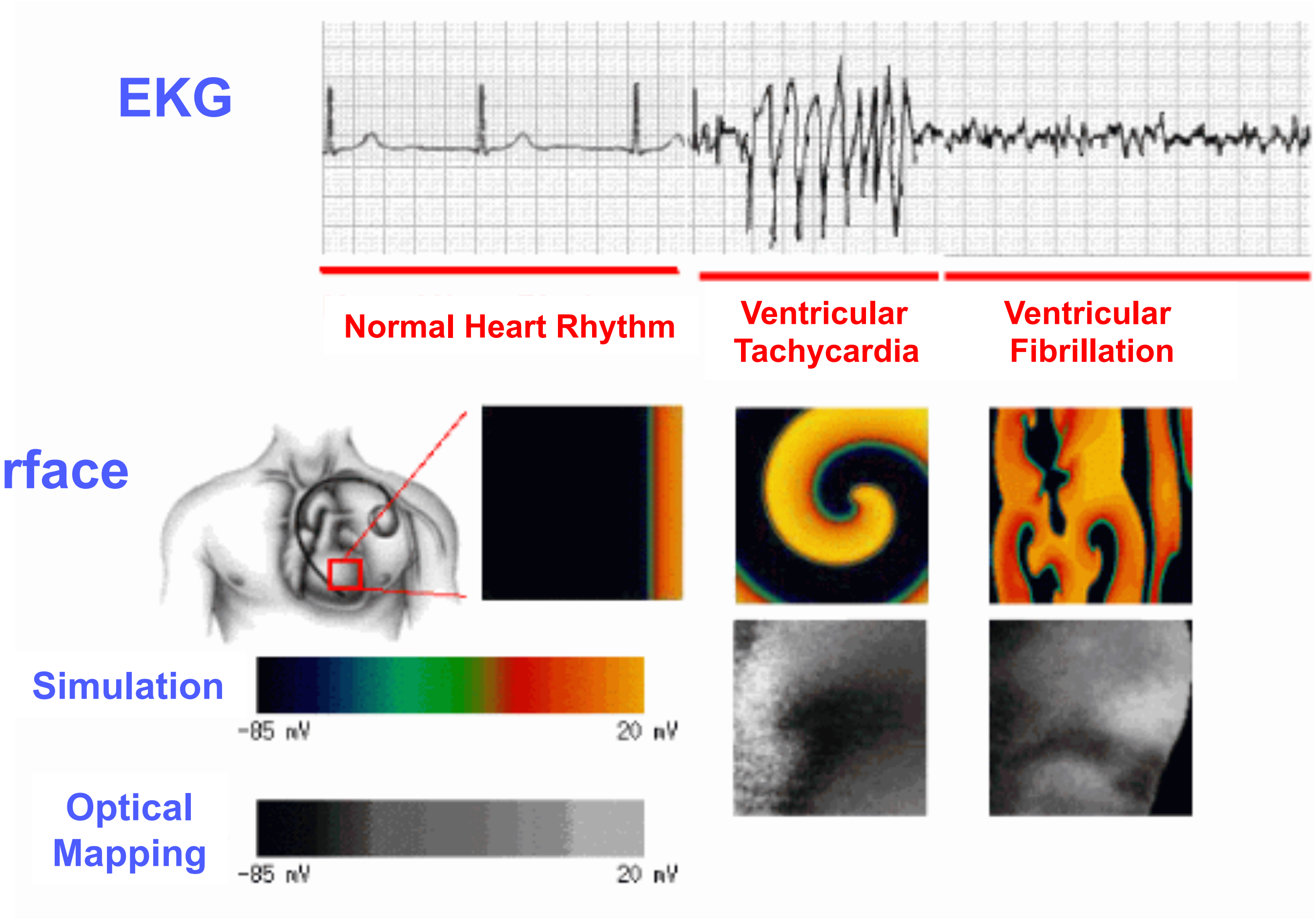
Outline

Modeling of hybrid systems

Analysis of hybrid systems

Appplication to cardiac dynamics

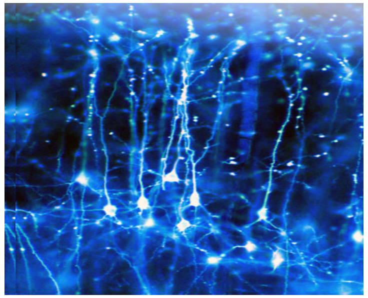
Emergent Behavior in Cardiac Cells



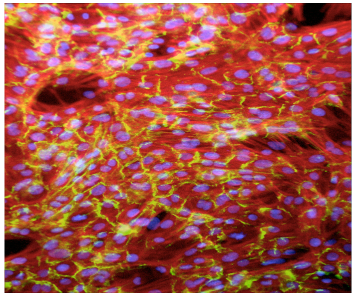
Arrhythmia afflicts more than 3 million Americans alone

Excitable Cells

- **Generate action potentials (AP),** electrical pulses in response to electric stimulation
  - **Examples:** neurons, cardiac cells, etc.
- **Local regeneration** allows electric signal propagation without damping
- **Building block** for electrical signaling in brain, heart, and muscles



Neurons of a squirrel
University College London



Artificial cardiac tissue
University of Washington

Single Cell Reaction: Action Potential

Membrane’s AP depends on:

- **Stimulus** (voltage or current):
  - External / Neighboring cells
- **Cell itself** (excitable or not):
  - State / Parameters value

Tissue: Reaction / diffusion

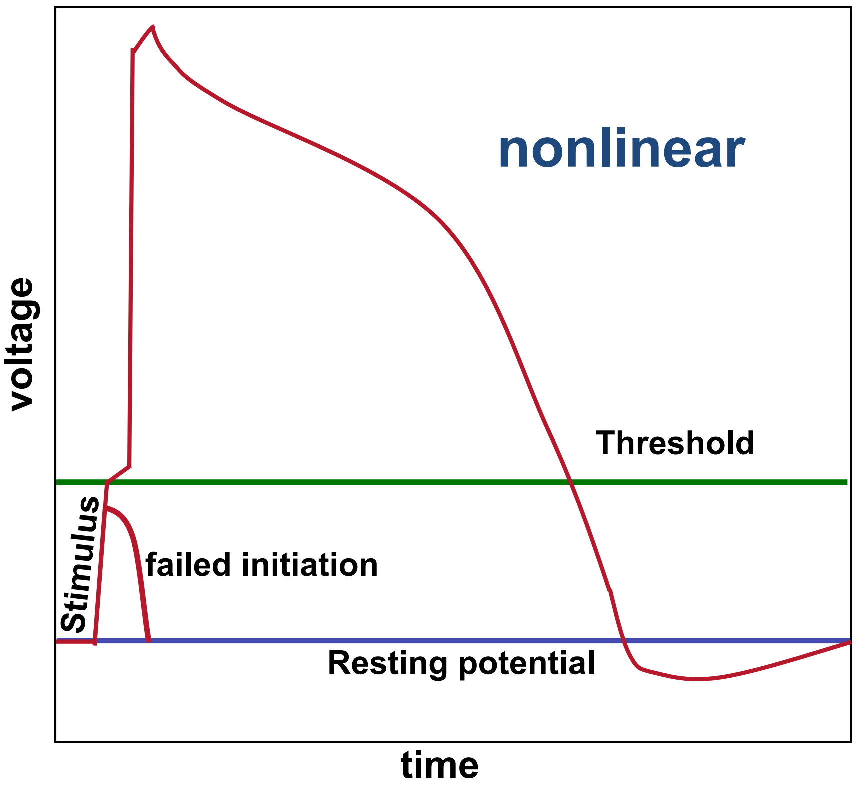
$$\frac{\partial \mathbf{u}}{\partial t} = R(\mathbf{u}) + \nabla(D\nabla\mathbf{u})$$

Behavior In time

Reaction

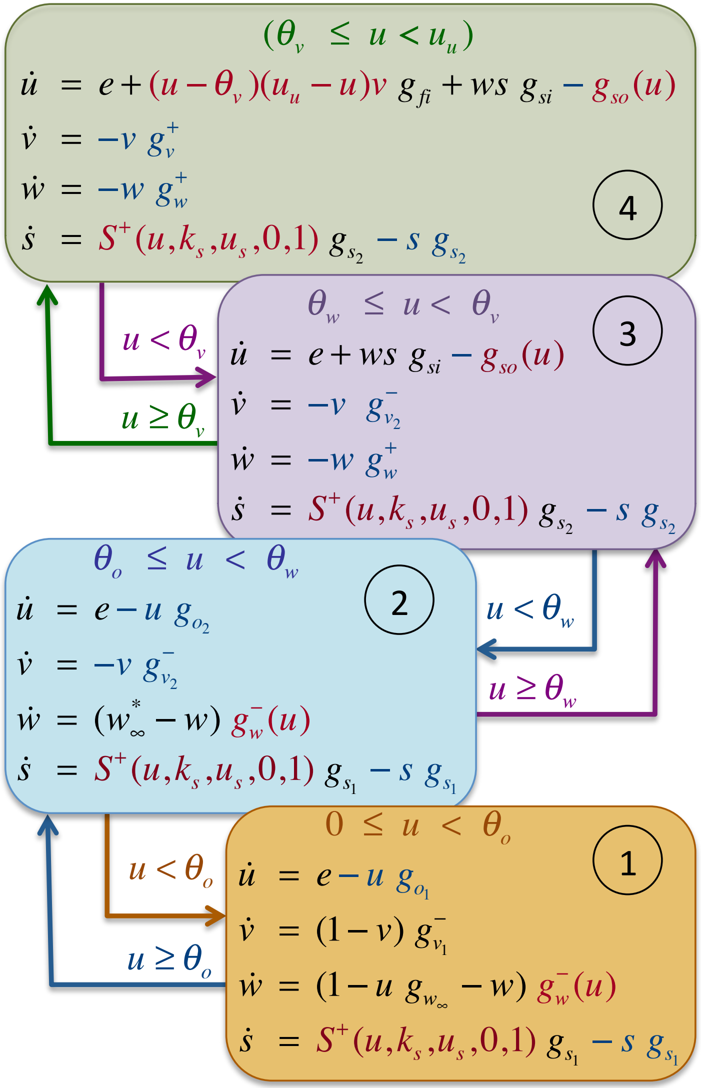
Diffusion

Schematic Action Potential



Minimal Model

Minimal Model as a Nonlinear Hybrid Automaton



④ $(\theta_v \le u < u_u)$

$$\dot{u} = e + (u - \theta_v)(u_u - u)v\, g_{fi} + ws\, g_{si} - g_{so}(u)$$
$$\dot{v} = -v\, g_v^+$$
$$\dot{w} = -w\, g_w^+$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\, g_{s_2} - s\, g_{s_2}$$

③ $\theta_w \le u < \theta_v$

$$\dot{u} = e + ws\, g_{si} - g_{so}(u)$$
$$\dot{v} = -v\, g_{v_2}^-$$
$$\dot{w} = -w\, g_w^+$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\, g_{s_2} - s\, g_{s_2}$$

② $\theta_o \le u < \theta_w$

$$\dot{u} = e - u\, g_{o_2}$$
$$\dot{v} = -v\, g_{v_2}^-$$
$$\dot{w} = (w_\infty^* - w)\, g_w^-(u)$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\, g_{s_1} - s\, g_{s_1}$$

① $0 \le u < \theta_o$

$$\dot{u} = e - u\, g_{o_1}$$
$$\dot{v} = (1 - v)\, g_{v_1}^-$$
$$\dot{w} = (1 - u\, g_{w_\infty} - w)\, g_w^-(u)$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\, g_{s_1} - s\, g_{s_1}$$

4 → 3: $u < \theta_v$

3 → 4: $u \ge \theta_v$

3 → 2: $u < \theta_w$

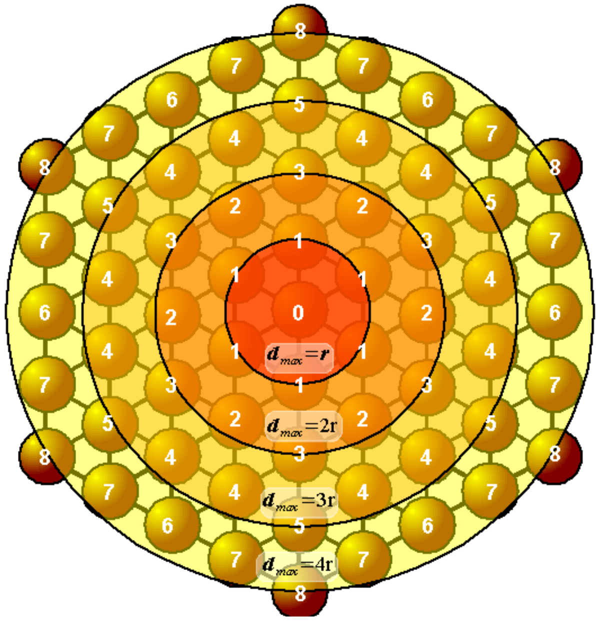
2 → 3: $u \ge \theta_w$

2 → 1: $u < \theta_o$

1 → 2: $u \ge \theta_o$

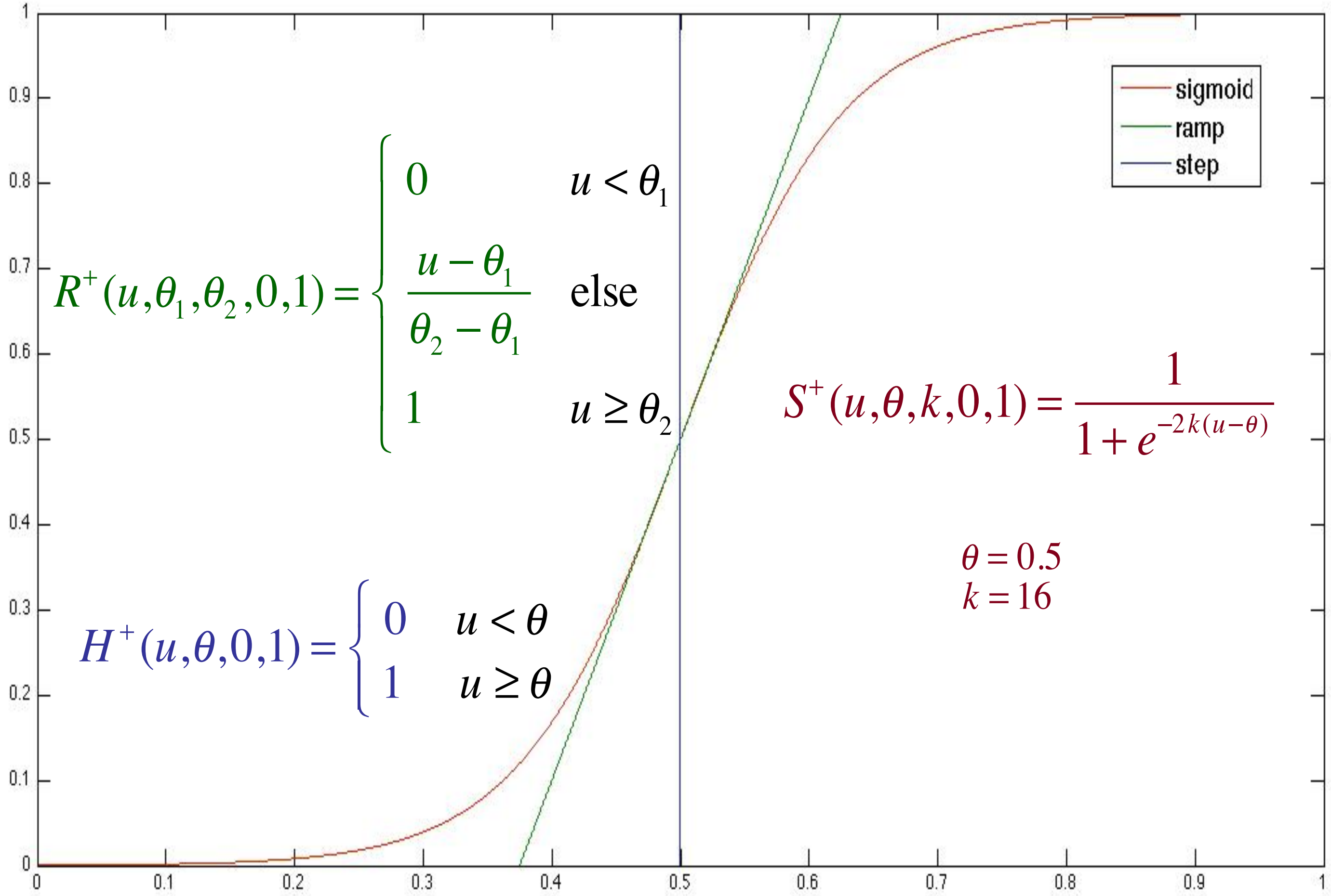
2D/3D Simulation of Partial Differential Equations

$$\dot{u} = \nabla(D\nabla u) - (J_{fi} + J_{si} + J_{so})$$



PDEs are simulated as Finite Difference Equations

Biological Switching



$$R^{+}(u,\theta_1,\theta_2,0,1)=\begin{cases}0 & u<\theta_1\\ \dfrac{u-\theta_1}{\theta_2-\theta_1} & \text{else}\\ 1 & u\geq\theta_2\end{cases}$$

$$S^{+}(u,\theta,k,0,1)=\frac{1}{1+e^{-2k(u-\theta)}}$$

$$H^{+}(u,\theta,0,1)=\begin{cases}0 & u<\theta\\ 1 & u\geq\theta\end{cases}$$

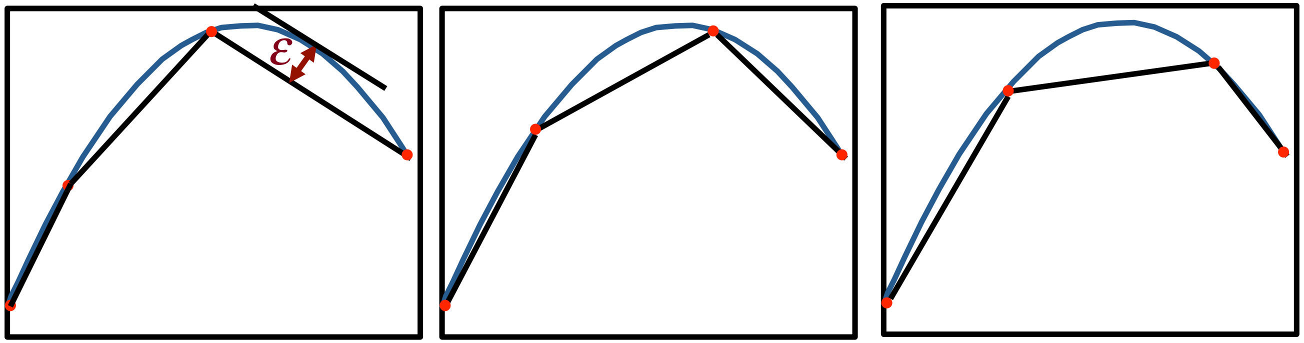
$\theta=0.5$
$k=16$

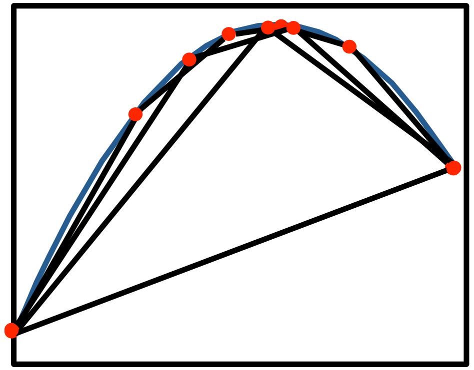
Optimal Polygonal Approximation

Given: One nonlinear curve and desired # segments

Find: Optimal polygonal approximation

- **Example: What is the optimal polygonal approximation of the blue curve with 3 segments?**





Dynamic Programming Algorithm

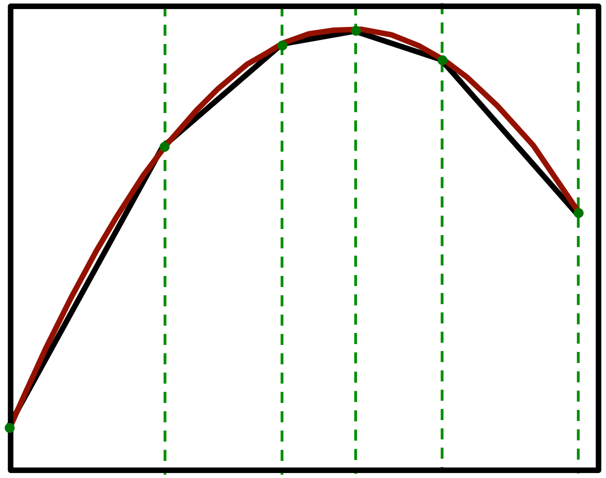
- **Complexity:** $O(P^2)$
- **P:** # points of the curve

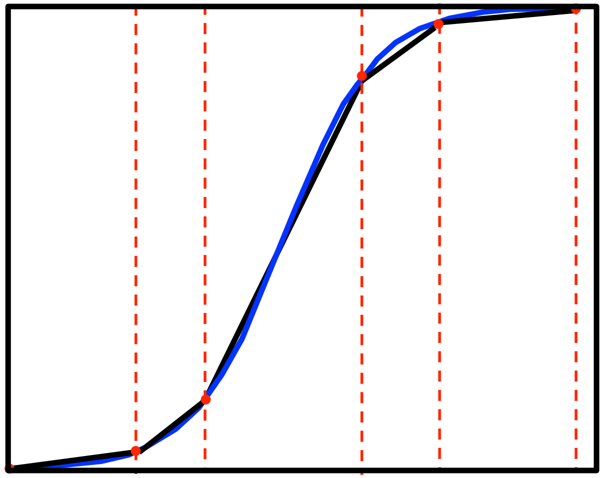
Globally-Optimal Polygonal Approximation

Given: Set of nonlinear curves and desired # of segments

Find: Globally optimal polygonal approximation

- **Example:** What is the optimal polygonal approximation of the curves below with 5 segments?



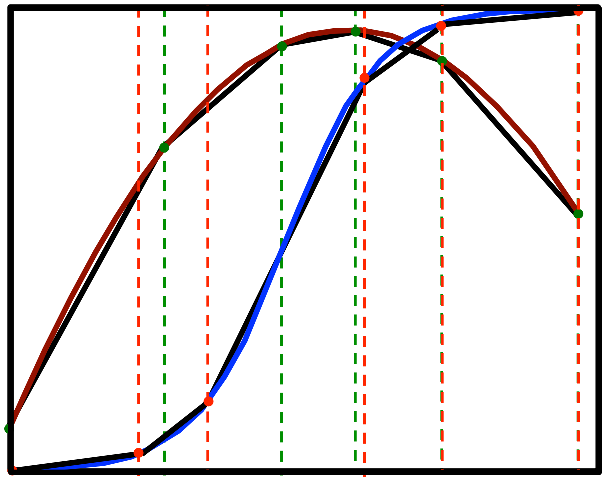


Globally-Optimal Polygonal Approximation

Given: Set of nonlinear curves and desired # of segments

Find: Globally optimal polygonal approximation

- **Example:** What is the optimal polygonal approximation of the curves below with 5 segments?



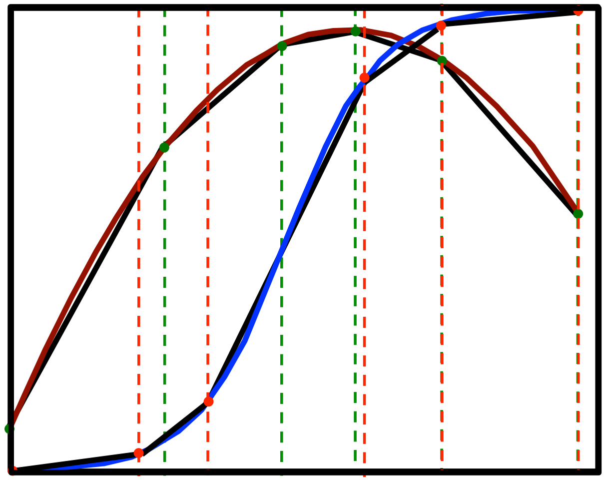
Combining the two we obtain 8 segments and not 5 segments

Globally-Optimal Polygonal Approximation

Given: Set of nonlinear curves and desired # of segments

Find: Globally optimal polygonal approximation

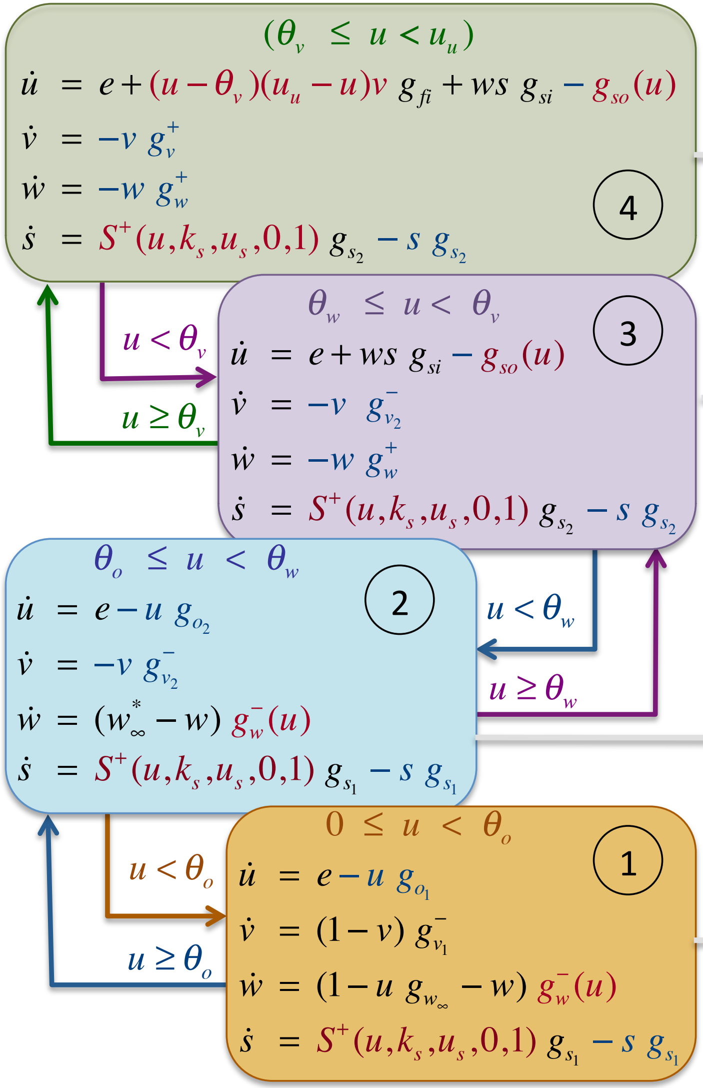
- **Example: What is the optimal polygonal approximation of the curves below with 5 segments?**



- **Solution: modify the OPAA to minimize the maximum error of a set of curves simultaneously.**

From MM Multi-Affine HA

Minimal Model as a Nonlinear Hybrid Automaton



④ $(\theta_v \le u < u_u)$

$$\dot{u} = e + (u - \theta_v)(u_u - u)v\; g_{fi} + ws\; g_{si} - g_{so}(u)$$
$$\dot{v} = -v\; g_v^+$$
$$\dot{w} = -w\; g_w^+$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\; g_{s_2} - s\; g_{s_2}$$

③ $\theta_w \le u < \theta_v$

$$\dot{u} = e + ws\; g_{si} - g_{so}(u)$$
$$\dot{v} = -v\; g_{v_2}^-$$
$$\dot{w} = -w\; g_w^+$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\; g_{s_2} - s\; g_{s_2}$$

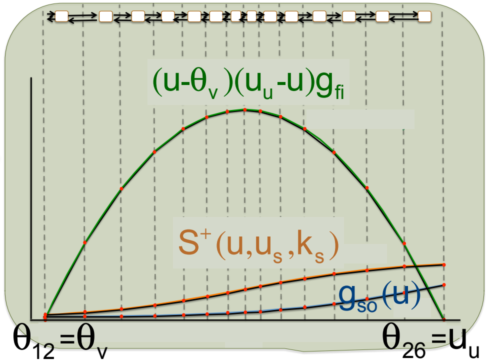
② $\theta_o \le u < \theta_w$

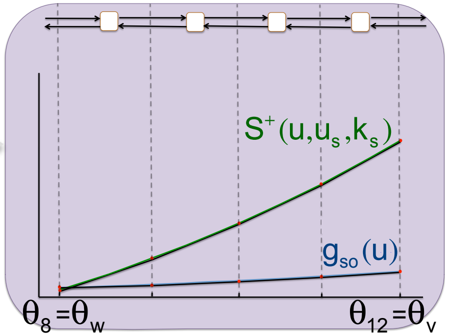
$$\dot{u} = e - u\; g_{o_2}$$
$$\dot{v} = -v\; g_{v_2}^-$$
$$\dot{w} = (w_\infty^* - w)\; g_w^-(u)$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\; g_{s_1} - s\; g_{s_1}$$

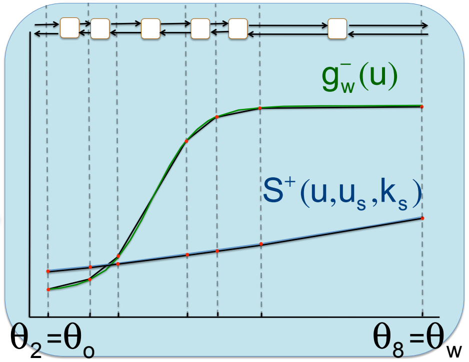
① $0 \le u < \theta_o$

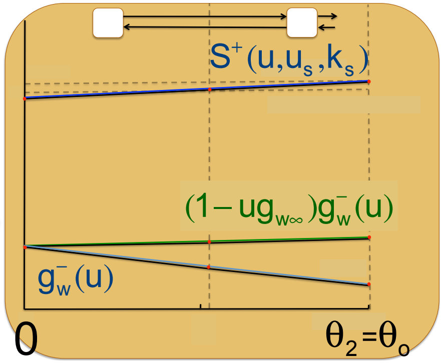
$$\dot{u} = e - u\; g_{o_1}$$
$$\dot{v} = (1 - v)\; g_{v_1}^-$$
$$\dot{w} = (1 - u\; g_{w_\infty} - w)\; g_w^-(u)$$
$$\dot{s} = S^+(u,k_s,u_s,0,1)\; g_{s_1} - s\; g_{s_1}$$

Optimal Linearization of Nonlinear Terms



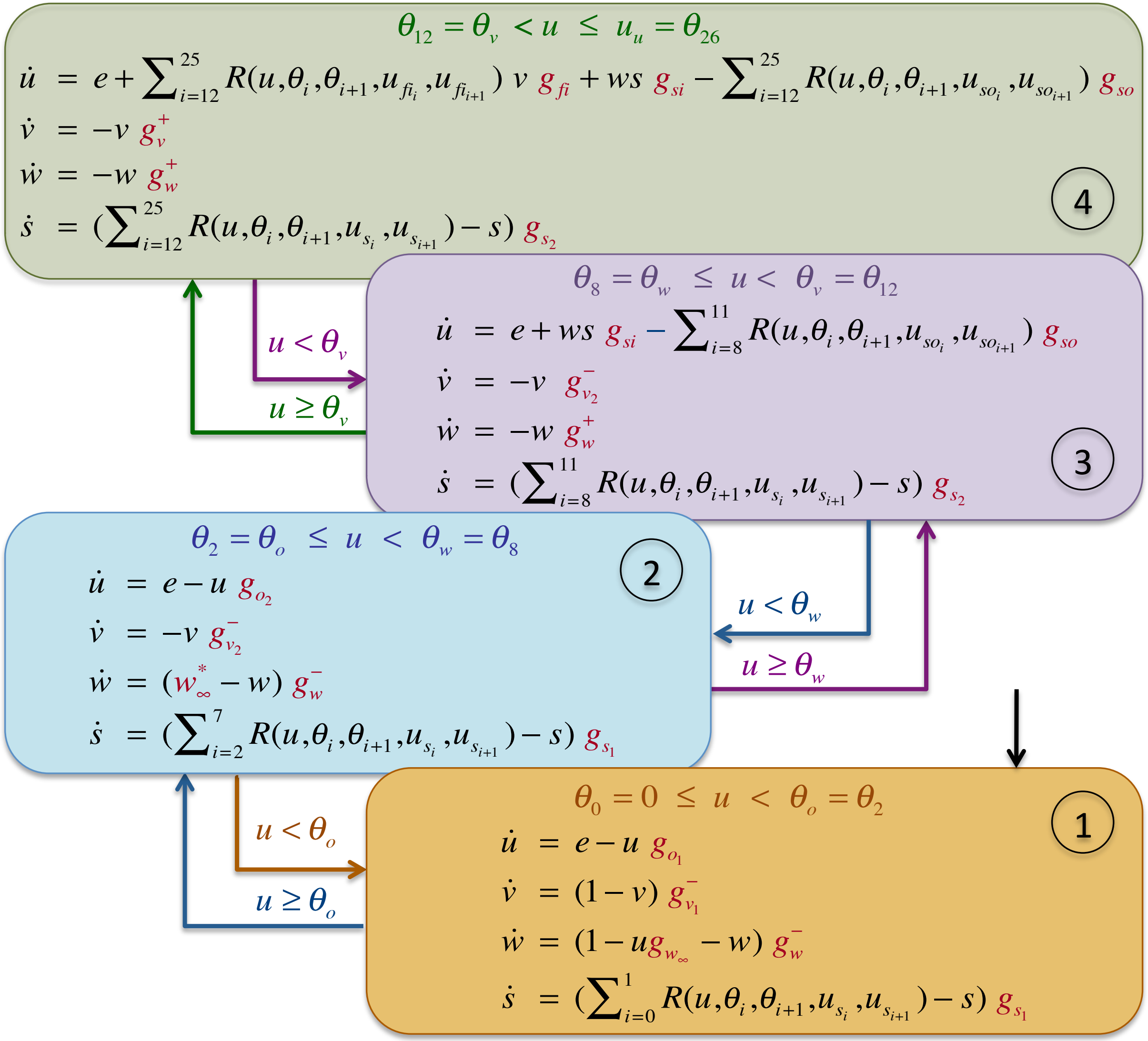






Validation of the Multi-Affine HA

The Multi-Affine Hybrid Automaton



$$\theta_{12} = \theta_v < u \le u_u = \theta_{26}$$

$$\begin{aligned}
\dot u &= e + \sum_{i=12}^{25} R(u,\theta_i,\theta_{i+1},u_{fi_i},u_{fi_{i+1}})\, v\, g_{fi} + ws\, g_{si} - \sum_{i=12}^{25} R(u,\theta_i,\theta_{i+1},u_{so_i},u_{so_{i+1}})\, g_{so} \\
\dot v &= -v\, g_v^+ \\
\dot w &= -w\, g_w^+ \\
\dot s &= (\sum_{i=12}^{25} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\, g_{s_2}
\end{aligned}$$

4

$u < \theta_v$

$u \ge \theta_v$

$$\theta_8 = \theta_w \le u < \theta_v = \theta_{12}$$

$$\begin{aligned}
\dot u &= e + ws\, g_{si} - \sum_{i=8}^{11} R(u,\theta_i,\theta_{i+1},u_{so_i},u_{so_{i+1}})\, g_{so} \\
\dot v &= -v\, g_{v_2}^- \\
\dot w &= -w\, g_w^+ \\
\dot s &= (\sum_{i=8}^{11} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\, g_{s_2}
\end{aligned}$$

3

$u < \theta_w$

$u \ge \theta_w$

$$\theta_2 = \theta_o \le u < \theta_w = \theta_8$$

$$\begin{aligned}
\dot u &= e - u\, g_{o_2} \\
\dot v &= -v\, g_{v_2}^- \\
\dot w &= (w_\infty^* - w)\, g_w^- \\
\dot s &= (\sum_{i=2}^{7} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\, g_{s_1}
\end{aligned}$$

2

$u < \theta_o$

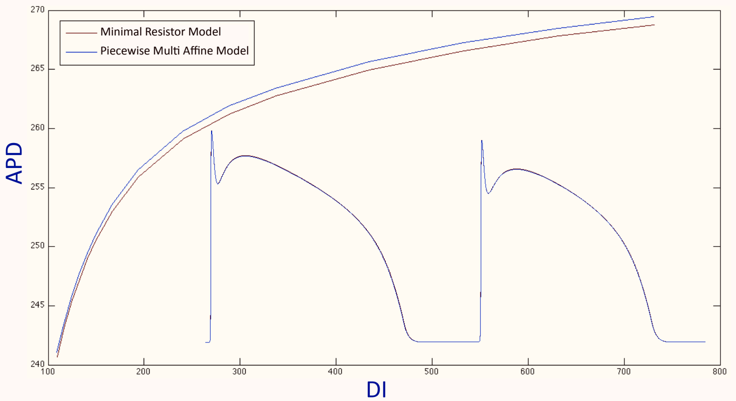
$u \ge \theta_o$

$$\theta_0 = 0 \le u < \theta_o = \theta_2$$

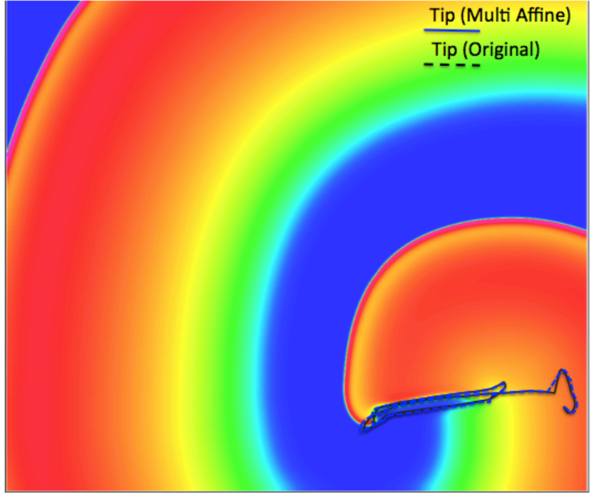
$$\begin{aligned}
\dot u &= e - u\, g_{o_1} \\
\dot v &= (1 - v)\, g_{v_1}^- \\
\dot w &= (1 - u g_{w_\infty} - w)\, g_w^- \\
\dot s &= (\sum_{i=0}^{1} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\, g_{s_1}
\end{aligned}$$

1

Comparison in 1D



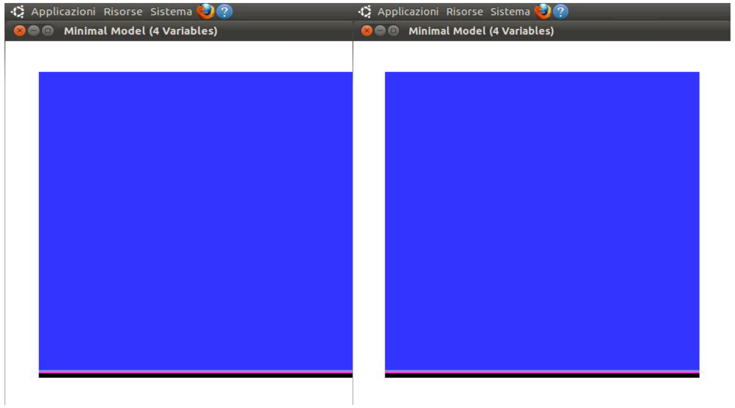
Comparison in 2D



Multi Affine (26 ramps)

Parameter-Synthesis Problem

Spiral Wave Induced by Unexcitable Myocytes



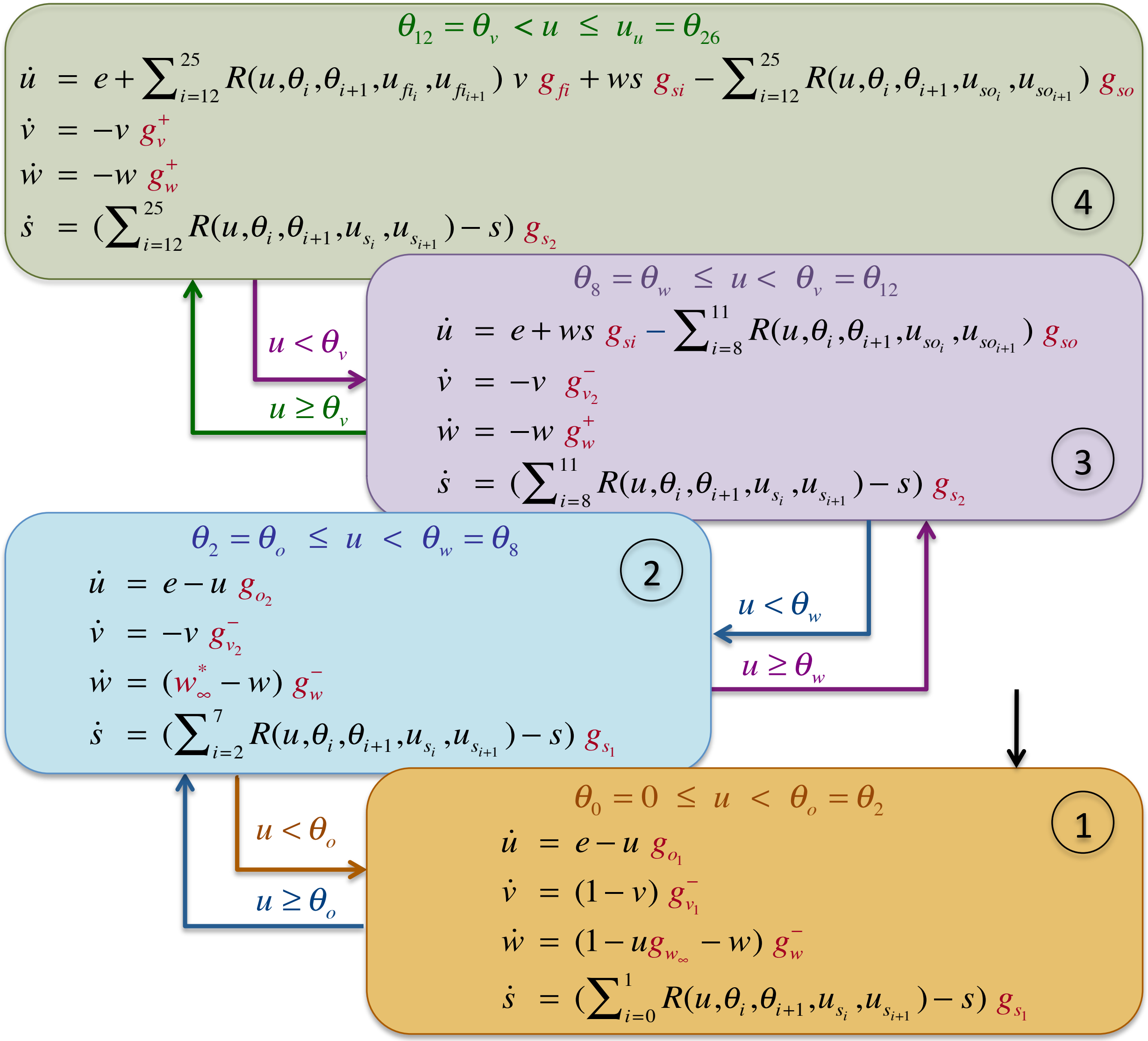
Property to Check

$$G(u < \theta_v)$$

Uncertain Parameters

$$g_{o_1} \in [0,180], \quad g_{o_2} \in [0,10]$$

$$g_{si} \in [0.1,100], \quad g_{so} \in [0.9,50]$$



$$\theta_{12} = \theta_v < u \leq u_u = \theta_{26}$$

$$\begin{aligned}
\dot{u} &= e + \sum_{i=12}^{25} R(u,\theta_i,\theta_{i+1},u_{fi_i},u_{fi_{i+1}})\ v\ g_{fi} + ws\ g_{si} - \sum_{i=12}^{25} R(u,\theta_i,\theta_{i+1},u_{so_i},u_{so_{i+1}})\ g_{so}\\
\dot{v} &= -v\ g_v^+\\
\dot{w} &= -w\ g_w^+\\
\dot{s} &= (\sum_{i=12}^{25} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\ g_{s_2}
\end{aligned}$$

$$\theta_8 = \theta_w \leq u < \theta_v = \theta_{12}$$

$$\begin{aligned}
\dot{u} &= e + ws\ g_{si} - \sum_{i=8}^{11} R(u,\theta_i,\theta_{i+1},u_{so_i},u_{so_{i+1}})\ g_{so}\\
\dot{v} &= -v\ g_{v_2}^-\\
\dot{w} &= -w\ g_w^+\\
\dot{s} &= (\sum_{i=8}^{11} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\ g_{s_2}
\end{aligned}$$

$$\theta_2 = \theta_o \leq u < \theta_w = \theta_8$$

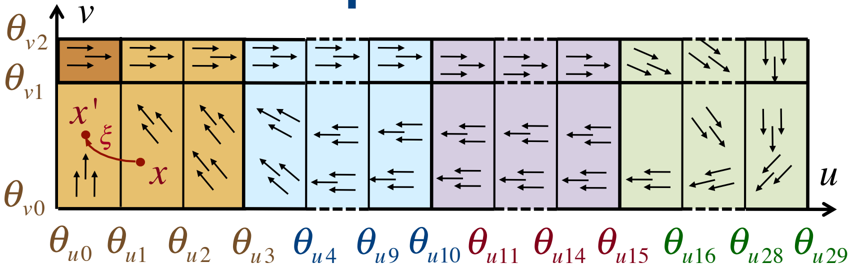
$$\begin{aligned}
\dot{u} &= e - u\ g_{o_2}\\
\dot{v} &= -v\ g_{v_2}^-\\
\dot{w} &= (w_\infty^* - w)\ g_w^-\\
\dot{s} &= (\sum_{i=2}^{7} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\ g_{s_1}
\end{aligned}$$

$$\theta_0 = 0 \leq u < \theta_o = \theta_2$$

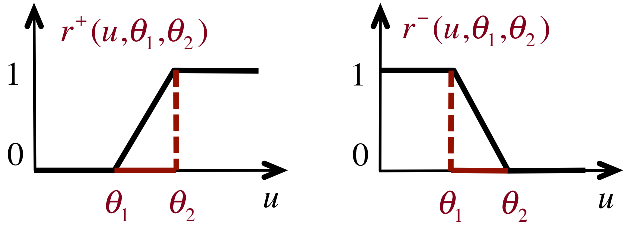
$$\begin{aligned}
\dot{u} &= e - u\ g_{o_1}\\
\dot{v} &= (1 - v)\ g_{v_1}^-\\
\dot{w} &= (1 - u g_{w_\infty} - w)\ g_w^-\\
\dot{s} &= (\sum_{i=0}^{1} R(u,\theta_i,\theta_{i+1},u_{s_i},u_{s_{i+1}}) - s)\ g_{s_1}
\end{aligned}$$

Finitary Discrete Abstraction

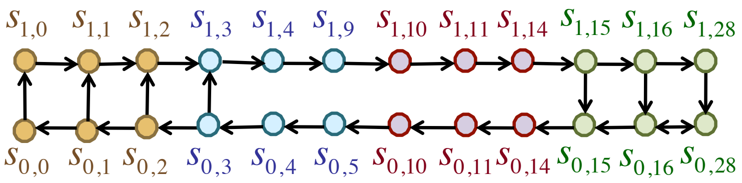
State-Space Partition



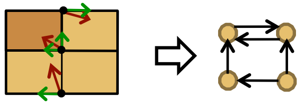
Ramps



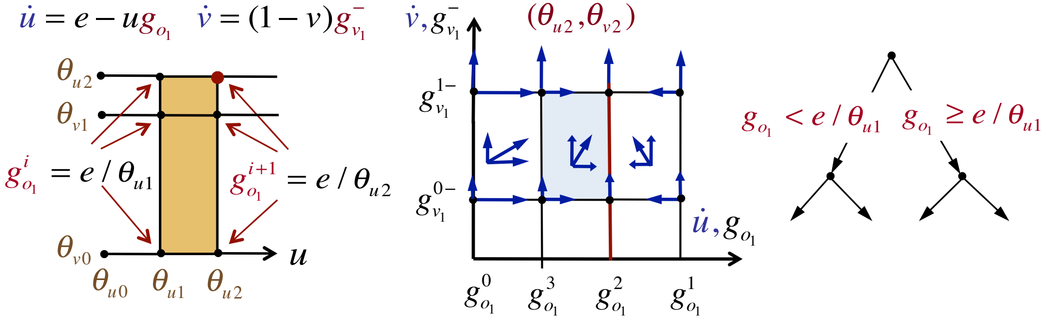
Kripke Structure for Fixed Parameters



Computation of transitions:
By examining corner flows

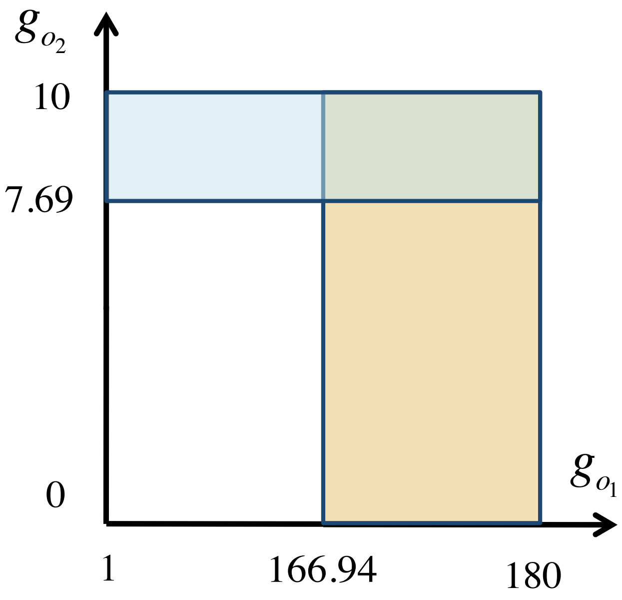


Parameter-Space Partition

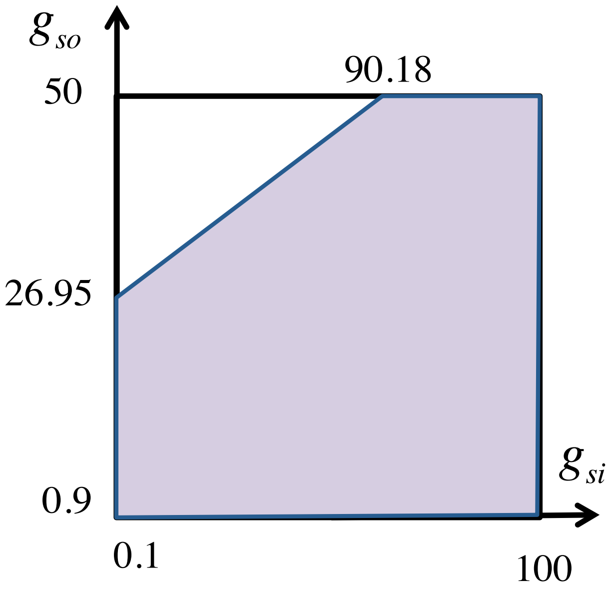


Results

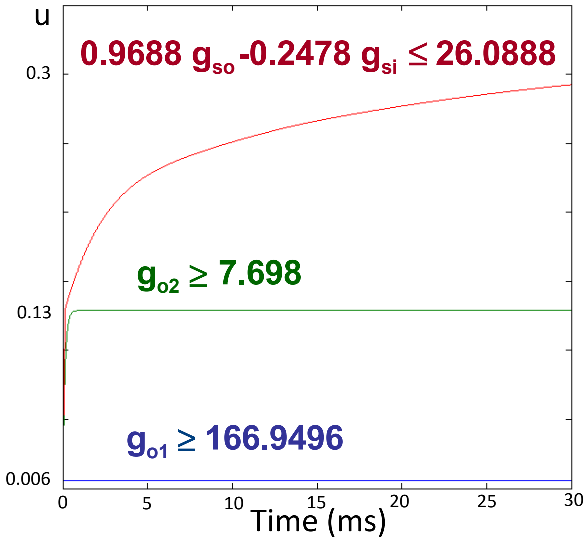
Rovergene: intelligently explores the **PS rectangles**



independent



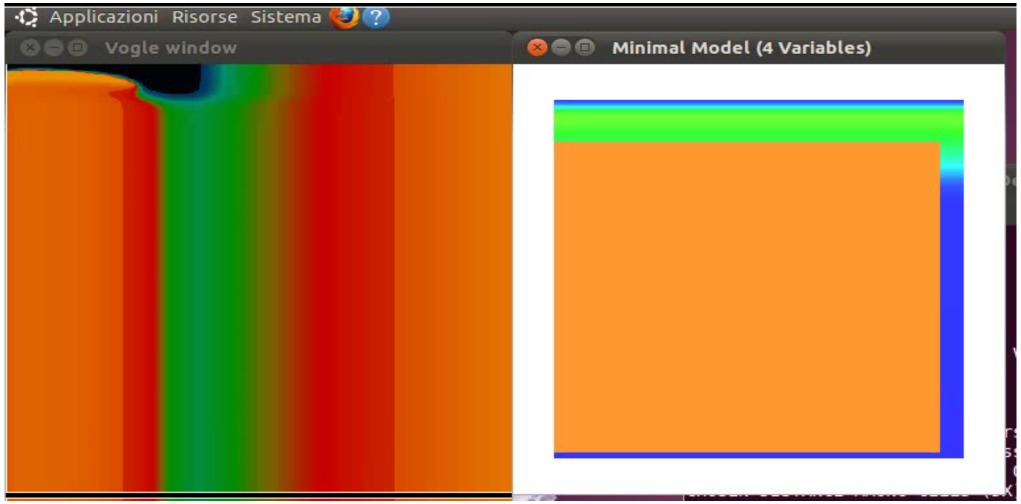
linearly dependent



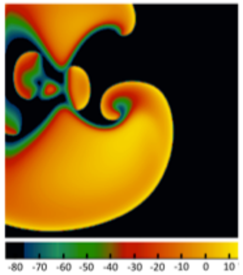
simulation

Analysis by Simulation

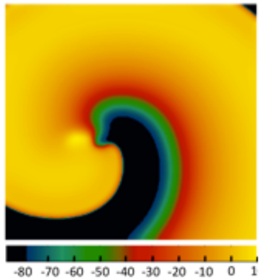
Minimal Model



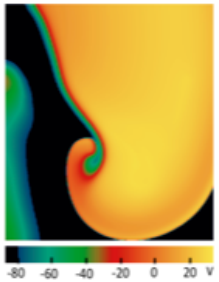
Beeler-Reuter (8 V)



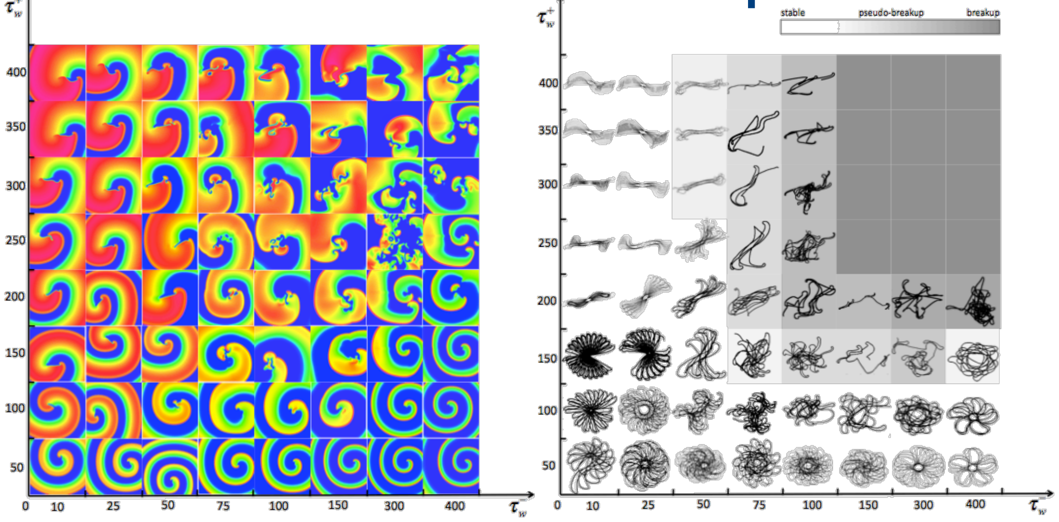
Ten-Tusscher-Panfilov (19V)

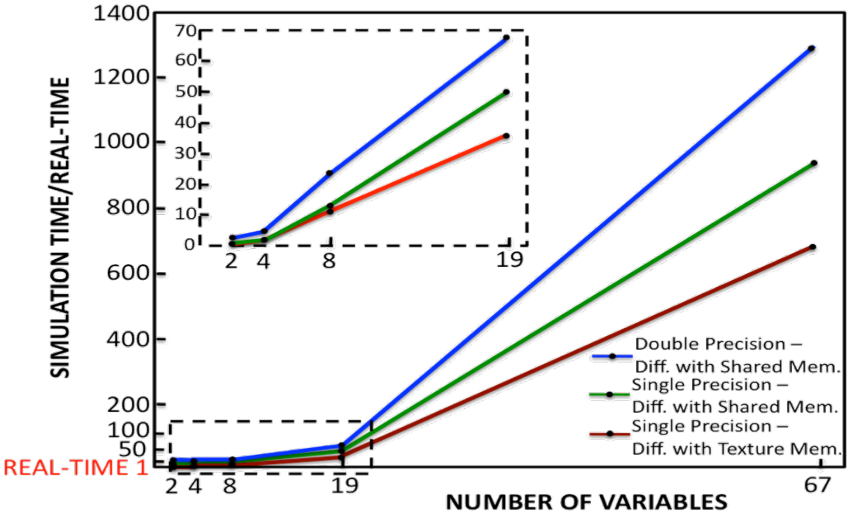


Iyer (65 V)



Spirals Computed by Students at CMACS Workshop

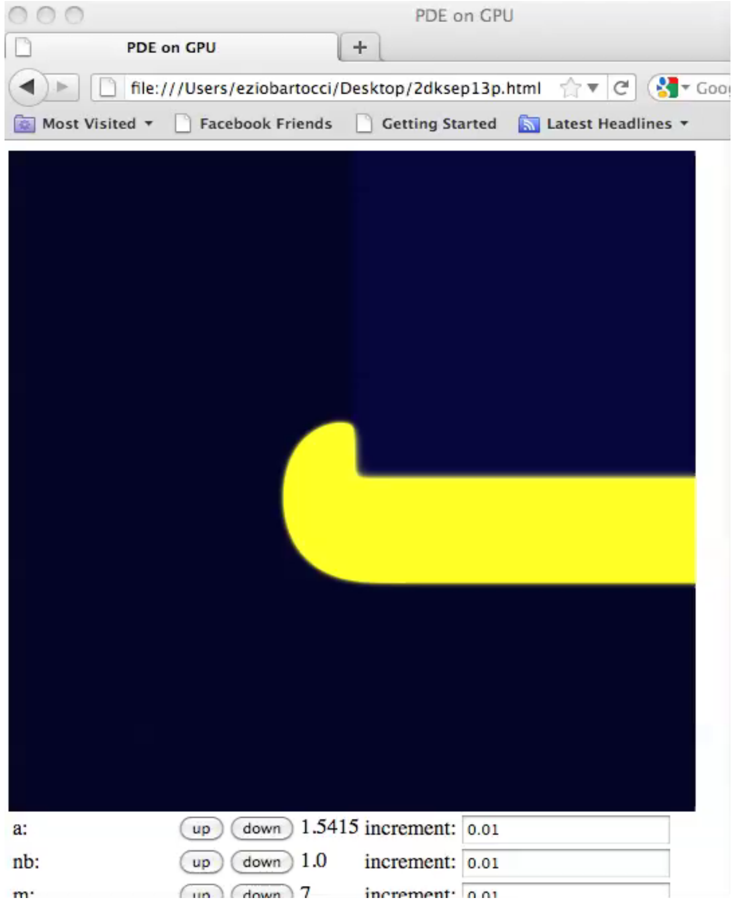




Simulation performance

3D-Models Simulation

Web Graphics Language (Fenton-Karma 2V)



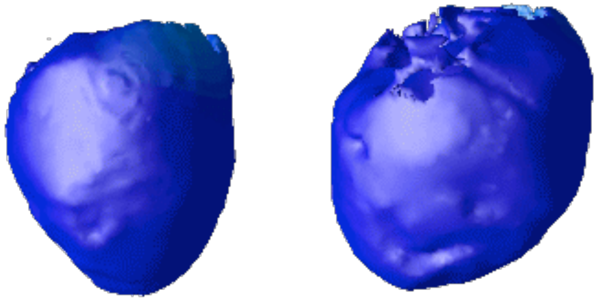
a: 1.5415 increment: 0.01

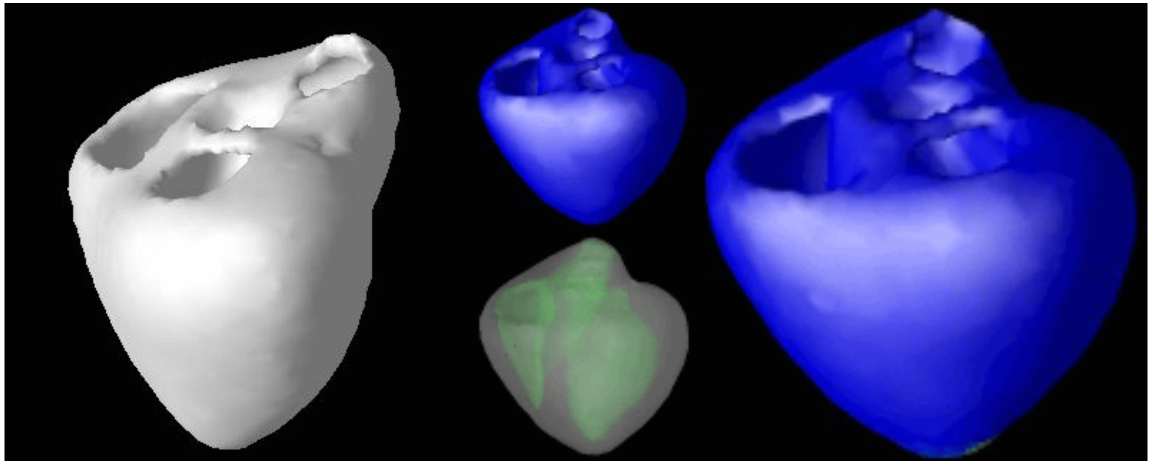
nb: 1.0 increment: 0.01

m: 7 increment: 0.01

Runs in your Browser and
Uses your own GPU

3D Model of a Mouse Heart (Fenton-Karma 3V Model)





3D Model of a Pig Heart
(Fenton-Karma 3V **Model)**

References

R. Alur and D. Dill. A theory of timed automata. Theoretical Computer Science 126:183 – 235, 1994 (prelim. versions app. in Proc. of 17th ICALP, LNCS 443, 1990, and Real Time: Theory in Practice, LNCS 600, 1991

R. Alur, C. Courcoubetis, N. Halbwachs, T.A. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, S. Yovine. The Algorithmic Analysis of Hybrid Systems. Theoretical Computer Science 138:3-34, 1995

T.A. Henzinger. The Theory of Hybrid Automata. Proceedings of LICS'96, the 11th Annual Symposium on Logic in Computer Science, IEEE Computer Society Press, pp. 278-292, 1996.

A. Chutinan and B.H. Krogh. Computing Polyhedral Approximations to Flow Pipes for Dynamic Systems. In CDC'98, the 37th IEEE Conference on Decision and control, pp. 2089 – 2095, IEEE Press, 1998.

References

R. Alur, T.A. Henzinger, G. Lafferriere, and G.J. Pappas. Discrete Abstractions of Hybrid Systems. Proceedings of the IEEE, 2000.

T.A. Henzinger and R. Majumdar. Symbolic Model Checking for Rectangular Hybrid Systems. In TACAS'00, the Proc. of the 6th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems, LNCS 1785, pp. 142 – 156, Springer, 2000.

R. Alur, R. Grosu, Y. Hur, V. Kumar, and I. Lee. Modular Specification of Hybrid Systems in CHARON. In Proc. of HSCC'00, the 3rd Int. Conf. on Hybrid Systems: Computation and Control, Pittsburgh, March, 2000, LNCS 179, pp. 6 – 19, Springer, 2000.

R. Alur, R. Grosu, I. Lee, O. Sokolsky. Compositional Refinement for Hierarchical Hybrid Systems. In Proc. of HSCC'01, the 4th International Conf. on Hybrid Systems: Computation and Control, Rome, Italy, March, 2001, pp. 33 – 49, Springer, LNCS 2034.

References

G. Batt, C. Belta and R. Weiss. Model Checking Genetic Regulatory Networks with Parameter Uncertainty. In Proc. of *HSCC'07, the 10th Int. Conf. on Hybrid Systems : Computation and Control*, Pisa, Italy, 2007.

P. Tabuada. Verification and Control of Hybrid Systems: A Symbolic Approach. Springer, 2009.

C. Le Guernic and A. Girard. Reachability Analysis of Linear Systems using Support Functions. Nonlinear Analysis: Hybrid Systems, 42(2):250 – 262, Electronic Edition, 2010.

C. Le Guernic and A. Girard. Reachability Analysis of Linear Systems using Support Functions. Nonlinear Analysis: Hybrid Systems, 42(2):250 – 262, Electronic Edition, 2010.

References

G. Frehse, C. Le Guernic, A. Donze, R. Ray, O. Lebeltel, R. Ripado, A. Girard, T. Dang, O. Maler. SpaceEx: Scalable Verication of Hybrid Systems. In Proc. of *CAV'11, The 23rd Int. Conf. on Computer Aided Verification*, Snowbird, USA, LNCS 6806, pp. 379 – 395, 2011.

R. Grosu, G. Batt, F. Fenton, J. Glimm, C. Le Guernic, S.A. Smolka and E. Bartocci. From Cardiac Cells to Genetic Regulatory Networks. In Proc. of CAV'11, the *23rd Int. Conf. on Computer Aided Verification*, Cliff Lodge, Snowbird, Utah, USA, July, 2011, pp. 396 – 411, Springer, LNCS 6806.

Verification Tools for Hybrid Systems

HyTech: LHA

http://embedded.eecs.berkeley.edu/research/hytech/

PHAVer: LHA + affine dynamics

http://www-verimag.imag.fr/~frehse/

d/dt: affine dynamics + controller synthesis

http://www-verimag.imag.fr/~tdang/Tool-ddt/ddt.html

Matisse Toolbox: zonotopes

http://www.seas.upenn.edu/~agirard/Software/MATISSE/

HSOLVER: nonlinear systems

http://hsolver.sourceforge.net/

SpaceEx: LHA + affine dynamics

http://spaceex.imag.fr/