Hybrid Systems and Systems Biology

Alberto Policriti

Dpt. of Mathematics and Informatics, University of Udine.

Applied Genomics Institute

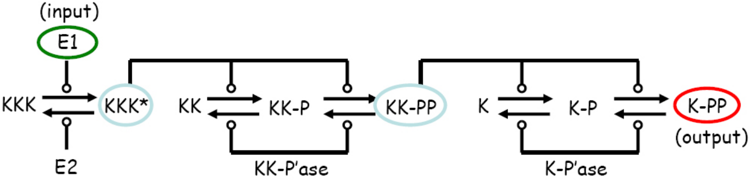
IGA ISTITUTO DI GENOMICA APPLICATA

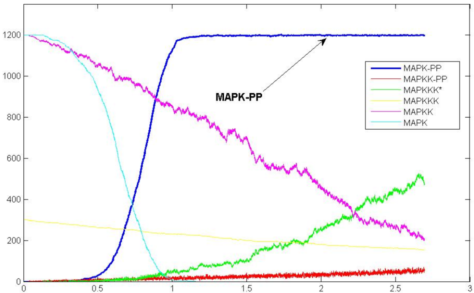
October 23rd, 2007

Outline

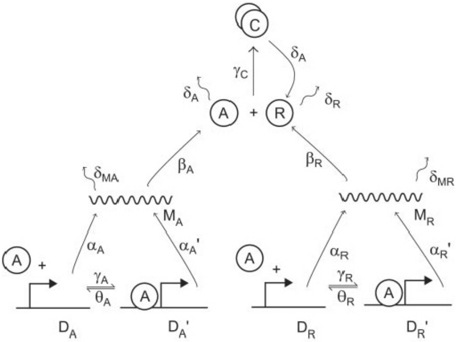
- Hybrid Systems: definition and applications to Systems Biology;
- a Stochastic Process Algebra (SPA) for biological modeling: *sCCP*;
- from sCCP to ODE's;
- problems and about ... being more "discrete".

A biological circuit: MAPKinase

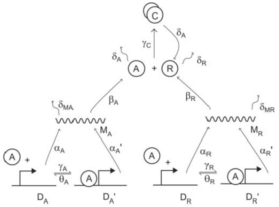




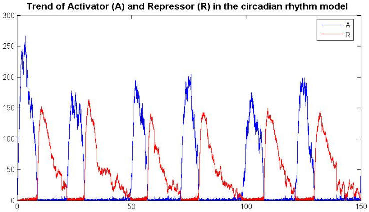
Circadian Clock (J. M. G. Vilar, H. Yuan Kueh, N. Barkai, and S. Leibler. PNAS, 2002.)

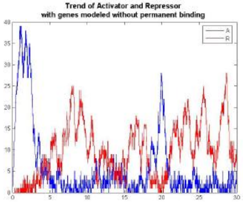


Circadian Clock



p_gate($\alpha_A, \alpha'_A, \gamma_A, \theta_A, M_A, A$) $\|$
p_gate($\alpha_R, \alpha'_R, \gamma_R, \theta_R, M_R, A$) $\|$
reaction(β_A, $[M_A]$, $[A]$) $\|$
reaction(δ_{MA}, $[M_A]$, $[]$) $\|$
reaction(β_R, $[M_R]$, $[R]$) $\|$
reaction(δ_{MR}, $[M_R]$, $[]$) $\|$
reaction(γ_C, $[A, R]$, $[AR]$) $\|$
reaction(δ_A, $[AR]$, $[R]$) $\|$
reaction(δ_A, $[A]$, $[]$) $\|$
reaction(δ_R, $[R]$, $[]$)





Modeling: Stochastic vs. Diff. Equations

Differential Equations

- mature
- computationally affordable (*one* run)

Stochastic *something*

- precise
- computationally costly (*many* runs!)

Hybrid Systems

Many real systems have a double nature. They:

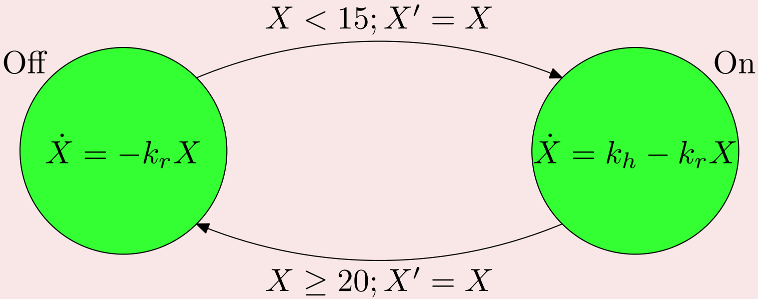
- evolve in a continuous way,
- are ruled by a discrete system.

Modeling?

hybrid systems/automata

The Example

A thermostat model



Hybrid Automata - Syntax

Alur et al. 1992

Definition (Hybrid Automaton - Syntax)

A tuple $H = \langle Z, Z', \mathcal{V}, \mathcal{E}, Inv, Dyn, Act, Reset \rangle$ where:

- Z and Z' are varibles in $\mathbb{R}^k$
- $\langle \mathcal{V}, \mathcal{E} \rangle$ is a graph
- Each $v \in \mathcal{V}$ is labelled by $Inv(v)[Z]$ and $Dyn(v)[Z, Z', T]$
- Each $e \in \mathcal{E}$ is labelled by $Act(e)[Z]$ and $Reset(e)[Z, Z']$

We consider $Dyn(v)[Z, Z', T]$ of the form $Z' = p_v(Z, T)$, where p_v is the solution of the vectorial field $\mathcal{P}(v)$.

Hybrid Automata - Intuitively

Finite Automata *plus Time*

Time flows when *within* states:

- H evolves from Z to Z' in time T when $Dyn(v)[Z, Z', T]$
- in *mode* v, Z must always satisfy $Inv(v)[Z]$
- H can cross e only if $Act(e)[Z]$
- when H crosses e, $Reset(e)[Z, Z']$

Hybrid Automata - States and Transitions

Definition (Hybrid Automaton State)

A state is a pair in $\mathcal{V} \times \mathbb{R}^k$.

Definition (Continuous Transition)

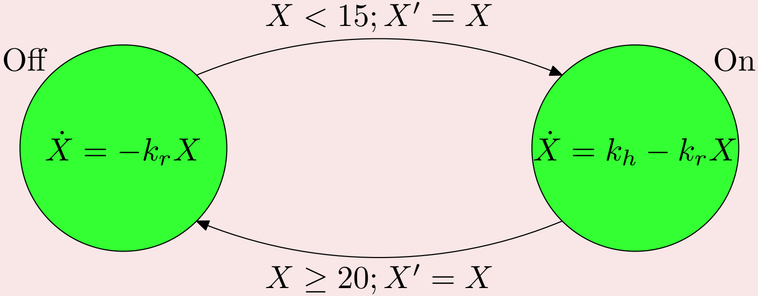
$\langle v, r\rangle \xrightarrow{t}_C \langle v, s\rangle \iff$ $\exists f : \mathbb{R}^+ \mapsto \mathbb{R}^k$ continuous such that $r = f(0)$, $s = f(t)$, and $\forall t' \in [0, t]$ the formulæ $Inv(v)[f(t')]$ and $Dyn(v)[r, f(t'), t']$ hold.

Definition (Discrete Transition)

$\langle v, r\rangle \xrightarrow{\langle v,u\rangle}_D \langle u, s\rangle \iff$ $\langle v, u\rangle \in \mathcal{E}$ and $Inv(v)[r]$, $Act(\langle v, u\rangle)[r]$, $Reset(\langle v, u\rangle)[r, s]$, and $Inv(u)[s]$ hold.

The Example

A thermostat model



Hybrid Automaton's Traces and Reachability

Definition (Hybrid Automaton Trace)

A trace $\ell_0 \ldots \ell_n$ from ℓ_0 to ℓ_n is a sequence of admissible states such that either $\ell_{i-1} \xrightarrow{t}_C \ell_i$ or $\ell_{i-1} \xrightarrow{e}_D \ell_i$ for all $i \in [1, n]$.

Definition (Hybrid Automaton Reachability)

$q \in \mathbb{R}^k$ is reachable from $p \in \mathbb{R}^k$ if there exists a trace from a state $\langle v, p \rangle$ to a state $\langle u, q \rangle$.

Reachability Problem

Reachability

Given two sets S and T, is there any $p \in S$ and $q \in T$ such that q is reachable from p?

Reachability Problem

Reachability

Given two sets S and T, is there any $p \in S$ and $q \in T$ such that q is reachable from p?

Reachability can be used to verify system's properties

ϕ is always false in $H \iff \mathrm{Sat}(\phi)$ is not reachable from H's initial states

Decidability of the Reachability Problem

Question

Can I say when T is reachable from S?

Decidability of the Reachability Problem

Question

Can I say when T is reachable from S?

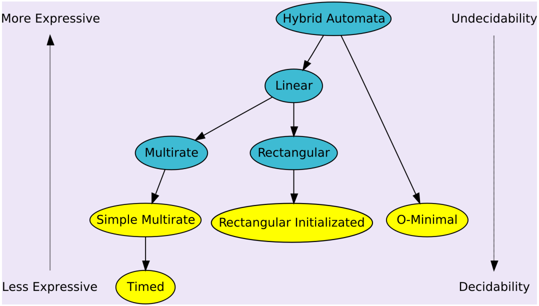
Answer

The *halting problem for 2-counters machines* can be reduced to a reachability problem over hybrid automata (Alur et al. 1995).

Decidability of the Reachability Problem

Question

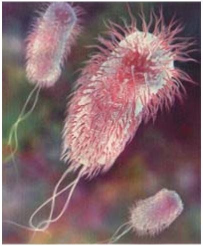
Can I say when T is reachable from S?



Examples of use of HS for Systems Biology

Escherichia coli

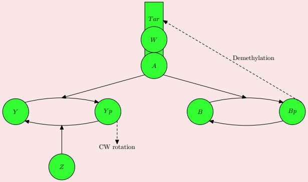
- a bacterium detecting the food concentration through a set of receptors;
- moving by flagellar rotations.



Depending on the concentration of attractans and repellents, E. coli responds to stimuli in one of two ways:

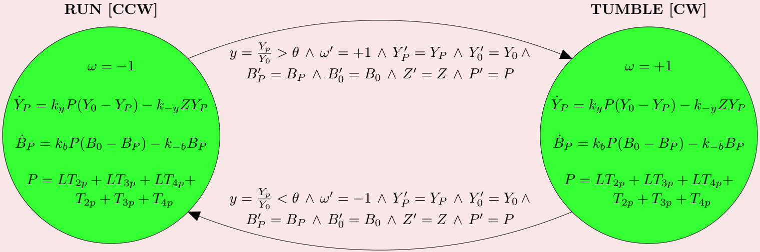
- "RUNS" – it moves in a straight line by moving its flagella counterclockwise (CCW)
- "TUMBLES" – it randomly changes its heading by moving its flagella clockwise (CW)

E. Coli Model



Examples of use of HS for Systems Biology

E. coli IDA Model



A. Casagrande et al., *Independent Dynamics Hybrid Automata in Systems Biology*, AB('05) Tokyo, 2005

Examples of use of HS for Systems Biology

Parameters in Genetic Regulatory Networks

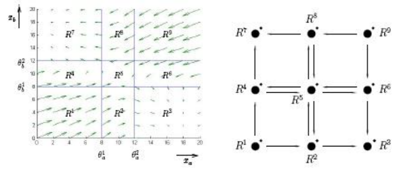
- Use "well behaving" differential equations (e.g. piece-wise multi affine functions);
- use temporal logic to express dynamical properties;
- partition the parameters'space in such a way to guarantee validity of temporal properties.
- PM-Systems *Model Checking Genetic Regulatory Networks with Applications to Synthetic Biology*, G. Batt and C. Belta

(Typical) Key Property

Theorem (Multiaffine functions on hyperrectangular polytopes) f multiaffine function and P hyperrectangular polytope:

$$f(P) \quad \subseteq \quad \text{hull}(\{f(v) \mid v \in \mathcal{V}_P\}),$$

that is $\forall x \in P$, $f(x)$ is a linear combination of the values of f at vertices of P.



Examples of use of HS for Systems Biology

Switching among simulation techniques

Use different simulation techniques as the number of molecule varies;

1. stochastic simulation for low numbers;
2. ode simulation for high numbers;

Alur et al. *Hybrid Modeling and Simulation of Biochemical Networks*

Modeling: Stochastic vs. Diff. Equations

Differential Equations

- mature
- computationally affordable (*one* run)

Stochastic *something*

- precise
- computationally costly (*many* runs!)

A *Bridge*: our attempt

First step

Use a Stochastic version of Concurrent Constraint Programming as Stochastic Process Algebra tool.

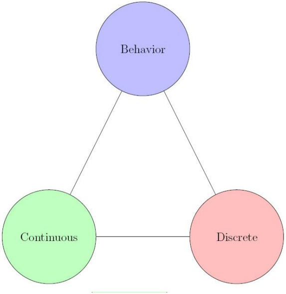
Problems! (with *behavioral* equivalence)

Second step

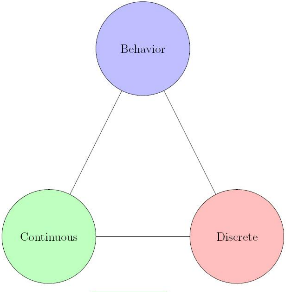
Introduce Hybrid Systems.

Modes of the HS $\Leftrightarrow$ Combinations of Stochastic choices
Dynamics $\Leftrightarrow$ *Ad-hoc* edge's variables with activations constrained by rates

The General View



The General View



Find the balance

1. Use the continuous simulation to decrease cost.
2. (Re)Introduce discrete transition to maintain behavioral equivalence.

Stochastic Concurrent Constraint Programming

What is

- A SPA with a computational "twist".
- Maintains a form of local storage.
- Keeps separated the description of interactions and the management of data for computations.
- (Naturally) Introduce functional rates.

Concurrent Constraint Programming

Constraint Store

- In this process algebra, the main objects are **constraints** , which are *formulae over an interpreted first order language* (i.e. $X = 10$, $Y > X - 3$).
- Constraints can be added to a "container", the constraint store, but can never be removed.

Agents

Agents can perform two basic operations on this store (asynchronously):

- Add a constraint (**tell** ask)
- Ask if a certain relation is entailed by the current configuration (**ask** instruction)

V. Saraswat, *Concurrent Constraint Programming*, MIT press, 1993

Syntax of CCP

$$Program = Decl.A$$

$$D = \varepsilon \mid Decl.Decl \mid p(x) : -A$$

$$\begin{array}{ll} A = & \mathbf{0} \\ \mid & \mathrm{tell(c).A} \\ \mid & \mathrm{ask(c_1).A_1 + ask(c_2).A_2} \\ \mid & A_1 \parallel A_2 \mid \exists_x A \mid p(x) \end{array}$$

Syntax of sCCP

Syntax of Stochastic CCP

$$Program = D.A$$

$$D = \varepsilon \mid D.D \mid p(\vec{x}) : -A$$

$$\begin{aligned} A &= \mathbf{0} \mid \mathrm{tell}_\infty(c).A \mid M \mid \exists_x A \mid A \parallel A \\ M &= \pi.G \mid M + M \\ \pi &= \mathrm{tell}_\lambda(c) \mid \mathrm{ask}_\lambda(c) \\ G &= \mathbf{0} \mid \mathrm{tell}_\infty(c).G \mid p(\vec{y}) \mid M \mid \exists_x G \mid G \parallel G \end{aligned}$$

L. Bortolussi, *Stochastic Concurrent Constraint Programming*, QAPL, 2006

Stochastic Rates

Rates are functions from the constraint store $\mathcal{C}$ to positive reals:

$$\lambda : \mathcal{C} \longrightarrow \mathbb{R}^+.$$

Rates can be thought as speed or duration of communications.

sCCP – technical details

Operational Semantics

▸ Show Details

- There are *two transition relations*, one instantaneous (finite and confluent) and one stochastic.
- Traces are sequences of events with variable time delays among them.

Discrete vs. Continuous Semantics

▸ Show Details

- The operational semantics is *abstract w.r.t. the notion of time*: we can map the labeled transition system into a discrete or a continuous time Markov Chain.

Implementation

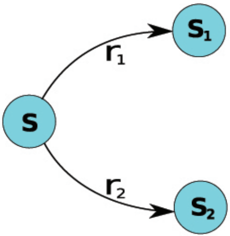
- We have an interpreter written in Prolog, using the *CLP engine of SICStus* to manage the constraint store.
- Efficiency issues.

Stream Variables

- *Quantities varying over time* can be represented in sCCP as unbounded lists.
- Hereafter: special meaning of $X = X + 1$.

Continuous Time Markov Chains

A **Continuous Time Markov Chain** (CTMC) is a direct graph with edges labeled by a real number, called the rate of the transition (representing the speed or the frequency at which the transition occurs).



- In each state, we select the next state according to a *probability distribution* obtained normalizing rates (from S to S_1 with prob. $\frac{r_1}{r_1+r_2}$).
- The time spent in a state is given by an exponentially distributed random variable, with rate given by the *sum of outgoing transitions* from the actual node ($r_1 + r_2$).

Biochemical Arrows to sCCP processes

$R_1 + \ldots + R_n \rightarrow_k P_1 + \ldots + P_m$	$\text{reaction}(k, [R_1, \ldots, R_n], [P_1, \ldots, P_m]) :-$ $\text{ask}_{r_{MA}(k, R_1, \ldots, R_n)} (\bigwedge_{i=1}^n (R_i > 0))$. $(\parallel_{i=i}^n \text{tell}_\infty(R_i = R_i - 1) \parallel_{j=1}^m \text{tell}_\infty(P_j = P_j + 1))$. $\text{reaction}(k, [R_1, \ldots, R_n], [P_1, \ldots, P_m])$
$R_1 + \ldots + R_n \rightleftharpoons_{k_2}^{k_1} P_1 + \ldots + P_m$	$\text{reaction}(k_1, [R_1, \ldots, R_n], [P_1, \ldots, P_m]) \parallel$ $\text{reaction}(k_2, [P_1, \ldots, P_m], [R_1, \ldots, R_n])$
$S \mapsto_{K, V_0}^{E} P$	$\text{mm\_reaction}(K, V_0, S, P) :-$ $\text{ask}_{r_{MM}(K, V_0, S)}(S > 0)$. $(\text{tell}_\infty(S = S - 1) \parallel tell_\infty(P = P + 1))$. $\text{mm\_reaction}(K, V_0, S, P)$
$S \mapsto_{K, V_0, h}^{E} P$	$\text{hill\_reaction}(K, V_0, h, S, P) :-$ $\text{ask}_{r_{Hill}(K, V_0, h, S)}(S > 0)$. $(\text{tell}_\infty(S = S - h) \parallel tell_\infty(P = P + h))$. $\text{Hill\_reaction}(K, V_0, h, S, P)$

where $r_{MA}(k, X_1, \ldots, X_n) = k \cdot X_1 \cdots X_n$; $r_{MM}(K, V_0, S) = \dfrac{V_0 S}{S + K}$; $r_{Hill}(k, V_0, h, S) = \dfrac{V_0 S^h}{S^h + K^h}$

Enzymatic reaction

$$S + E \rightleftharpoons_{k_{-1}}^{k_1} ES \rightarrow_{k_2} P + E$$

Mass Action Kinetics

enz_reaction($k_1, k_{-1}, k_2, S, E, ES, P$) :-
  reaction(k_1, [S, E], [ES]) ||
  reaction(k_{-1}, [ES], [E, S]) ||
  reaction(k_2, [ES], [E, P])

Mass Action Equations

$$\frac{d[ES]}{dt} = k_1[S][E] - k_2[ES] - k_{-1}[ES]$$
$$\frac{d[E]}{dt} = -k_1[S][E] + k_2[ES] + k_{-1}[ES]$$
$$\frac{d[S]}{dt} = -k_1[S][E]$$
$$\frac{d[P]}{dt} = k_2[ES]$$

Michaelis-Menten Equations

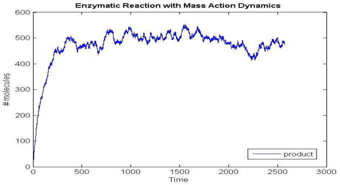
$$\frac{d[P]}{dt} = \frac{V_0 S}{S+K}$$
$$V_0 = k_2[E_0]$$
$$K = \frac{k_2+k_{-1}}{k_1}$$

Michaelis-Menten Kinetics

$$\text{mm\_reaction}\left(\frac{k_2 + k_{-1}}{k_1}, k_2 \cdot E, S, P\right)$$

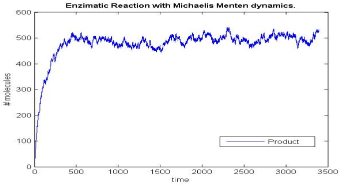
Enzymatic reaction

$$S + E \rightleftharpoons_{k_{-1}}^{k_1} ES \rightarrow_{k_2} P + E$$



Mass Action Kinetics

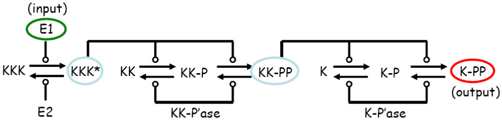
enz_reaction(k_1, k_{-1}, k_2, S, E, ES, P) :-
  reaction(k_1, [S, E], [ES]) ∥
  reaction(k_{-1}, [ES], [E, S]) ∥
  reaction(k_2, [ES], [E, P])



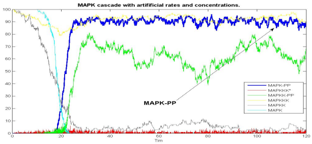
Michaelis-Menten Kinetics

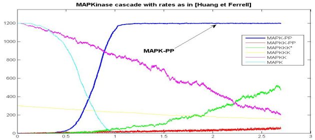
$$\mathrm{mm\_reaction}\left(\frac{k_2 + k_{-1}}{k_1}, k_2 \cdot E, S, P\right)$$

MAP-KINASE CASCADE

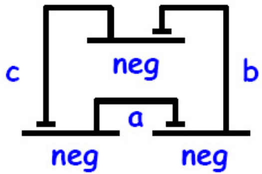


$$
\begin{array}{c}
enz\_reaction(k_a, k_d, k_r, KKK, E1, KKKE1, KKKS) \parallel \\
enz\_reaction(k_a, k_d, k_r, KKKS, E2, KKKSE2, KKK) \parallel \\
enz\_reaction(k_a, k_d, k_r, KK, KKKS, KKKKKS, KKP) \parallel \\
enz\_reaction(k_a, k_d, k_r, KKP, KKP1, KKPKKP1, KK) \parallel \\
enz\_reaction(k_a, k_d, k_r, KKP, KKKS, KKPKKKS, KKPP) \parallel \\
enz\_reaction(k_a, k_d, k_r, KP, KP1, KPKP1, K) \parallel \\
enz\_reaction(k_a, k_d, k_r, K, KKPP, KKKPP, KP) \parallel \\
enz\_reaction(k_a, k_d, k_r, KKPP, KKP1, KKPPKKP1, KKP) \parallel \\
enz\_reaction(k_a, k_d, k_r, KP, KKPP, KPKKPP, KPP) \parallel \\
enz\_reaction(k_a, k_d, k_r, KPP, KP1, KPPKP1, KP)
\end{array}
$$

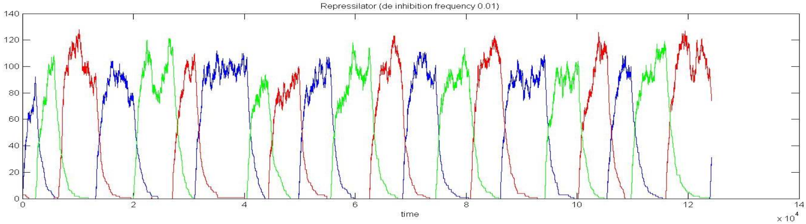




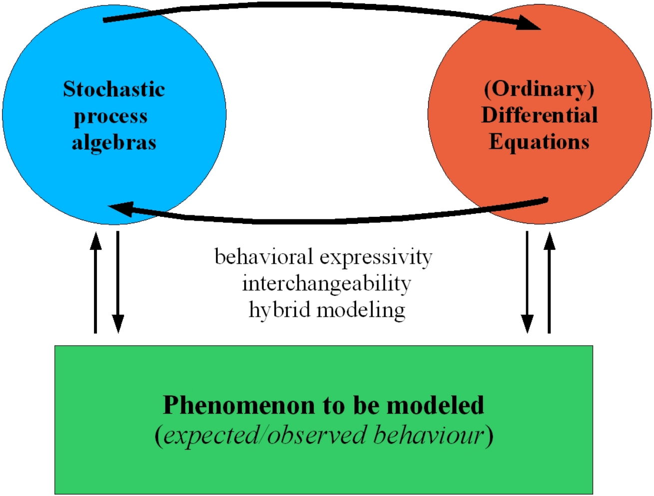
Repressilator



$\mathsf{neg\_gate}(0.1, 1, 0.0001, A, C) \parallel$
$\mathsf{reaction}(0.0001, [A], []) \parallel$
$\mathsf{neg\_gate}(0.1, 1, 0.0001, B, A) \parallel$
$\mathsf{reaction}(0.0001, [B], []) \parallel$
$\mathsf{neg\_gate}(0.1, 1, 0.0001, C, B) \parallel$
$\mathsf{reaction}(0.0001, [C], [])$



Connecting SPA and ODE models



From sCCP to ODE

What?

We want to associate a set of ODE to an sCCP program (written in a restricted syntax).

Why?

ODE can be numerically simulated faster than stochastic processes.

On the market...

There are (syntactic) methods to write set of ODEs for PEPA and stochastic π-calculus, looking at the speed of creation and destruction of terms (We did the same for sCCP).
However, the ODE can show a behavior different from that of SPA models.

From sCCP to ODE: example

Idea

Collapse all instantaneous transitions following a stochastic one and add their updates to the edge's label denoting such a transition.

Reduced Transitions Systems

- Associate a labeled graph to each sequential component of an sCCP program:

  EDGES are transitions and are labeled by a set of guards, a set of updates of variables of the store, and the corresponding rates;

  NODES are stochastic choices.

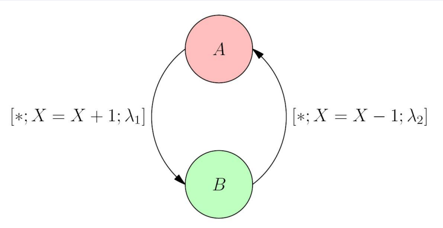
- Procedure calls are resolved by inserting a copy of the called procedure.
- Syntactic restrictions are necessaries.

From sCCP to ODE: example

Example

A :- $\mathsf{ask}_{\lambda_1}(\mathit{true}).\mathsf{tell}_{\infty}(X = X + 1)$.B
B :- $\mathsf{tell}_{\lambda_2}(X = X - 1)$.A

The RTS



From sCCP to ODE: example

Interaction matrix andx reaction vector

$$I = \begin{array}{c|cc} & t_1 & t_2 \\ \hline X & 1 & -1 \\ A & -1 & 1 \\ B & 1 & -1 \end{array} \qquad r = \begin{pmatrix} \lambda_1 \cdot A \\ \lambda_2 \cdot B \end{pmatrix}$$

$ode = I \cdot r$

$$ode \begin{cases} \dot{X} = \lambda_1 \cdot A - \lambda_2 \cdot B \\ \dot{A} = -\lambda_1 \cdot A + \lambda_2 \cdot B \\ \dot{B} = \lambda_1 \cdot A - \lambda_2 \cdot B \end{cases}$$

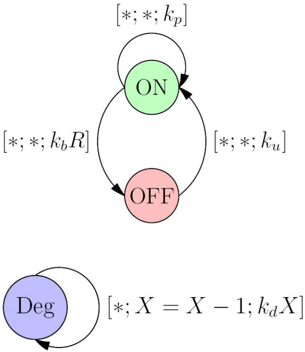
The strange beast of repressilator

Modeling 3 Negative Gene Gates

$$\mathrm{Neg}(X,R) \;\text{:-}\; \mathrm{tell}_{k_p}(X = X+1).\mathrm{Neg}(X,R) + \mathrm{ask}_{k_b R}(R \geq 1).\mathrm{ask}_{k_u}(\mathit{true}).\mathrm{Neg}(X,R)$$

$$\mathrm{Degrade}(X) \;\text{:-}\; \mathrm{ask}_{k_d X}(X > 0).\mathrm{tell}_{\infty}(X = X-1).\mathrm{Degrade}(X)$$

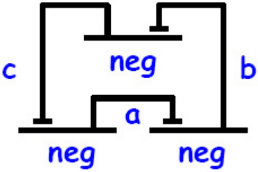
$$\mathrm{Neg}(A,C) \parallel \mathrm{Neg}(B,A) \parallel \mathrm{Neg}(C,B) \parallel \mathrm{Degrade}(A) \parallel \mathrm{Degrade}(B) \parallel \mathrm{Degrade}(C)$$



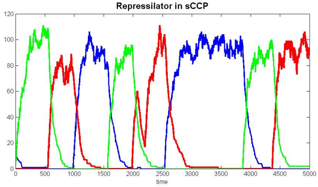
$$
\begin{aligned}
\dot{A} &= k_p Y_A - k_d A\\
\dot{B} &= k_p Y_B - k_d B\\
\dot{C} &= k_p Y_C - k_d C\\
\dot{Y}_A &= k_u Z_A - k_b Y_A C\\
\dot{Y}_B &= k_u Z_B - k_b Y_B A\\
\dot{Y}_C &= k_u Z_C - k_b Y_C B\\
\dot{Z}_A &= k_b Y_A C - k_u Z_A\\
\dot{Z}_B &= k_b Y_B A - k_u Z_B\\
\dot{Z}_C &= k_b Y_C B - k_u Z_C
\end{aligned}
$$

The strange beast of repressilator

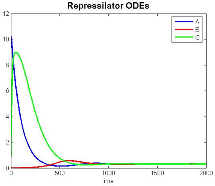
Repressilator with gene gates



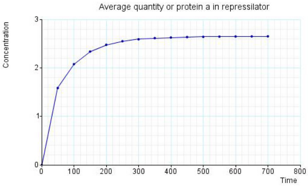
Repressilator in sCCP



Repressilator: ODE from sCCP



Repressilator: average of sCCP model



From sCCP to Hybrid Automata

What?

We want to associate an hybrid system to a sCCP network.

Why?

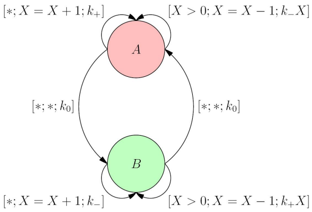
- The mixed discrete/continuous dynamics of HS is more natural than simple ODE, as it can preserve *some logical structure* of sCCP networks.
- Hybrid automata are equipped with well developed analysis methods.

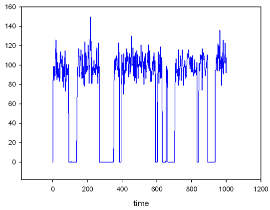
How?

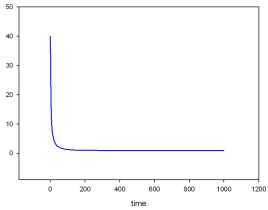
- The separation between constraint store and logical description of agents makes easy to identify (discrete) modes of the automata
- Activation conditions need to look at the temporal semantics of stochastic actions.

Example: a "distilled" Repressilator

$$
\begin{aligned}
A :- \quad & \mathsf{tell}_{k_+}(X = X + 1).A \\
+ \quad & \mathsf{ask}_{k_- X}(X > 0). \\
& \qquad \mathsf{tell}_{\infty}(X = X - 1).A \\
+ \quad & \mathsf{ask}_{k_0}(\mathit{true}).B \\
\\
B :- \quad & \mathsf{tell}_{k_-}(X = X + 1).B \\
+ \quad & \mathsf{ask}_{k_+ X}(X > 0). \\
& \qquad \mathsf{tell}_{\infty}(X = X - 1).B \\
+ \quad & \mathsf{ask}_{k_0}(\mathit{true}).A
\end{aligned}
$$







HA associated to an sCCP-ntwrk

Ideas

- localize the construction to looping edges in order to determine flow conditions;
- use (non constant) rates to govern variables associated to edges;
- use variables associated to edges in activation conditions.

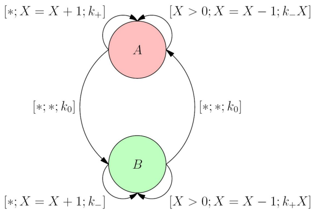
HA associated to an sCCP-ntwrk

$N = A_1 \parallel \ldots \parallel A_M$ be an sCCP-network.

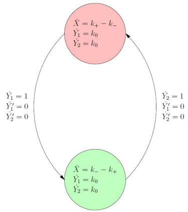
Definition (sketch)

1. *control modes* $\Sigma = (\sigma_1, \ldots, \sigma_M)$;
2. *control edges* corresponding to non-looping arcs $t_{ij} \in T_i$ of $RTS(A_i)$;
3. *variables*: stream variables $X_1, \ldots, X_k$ of N, plus one variable $Y_{i,j}$ for each RTS-edge t_{ij};
4. *flow conditions* $ode_\Sigma = \sum_{i=1}^{M} ode_{i,\sigma_i}$, where $ode_{i,\sigma_i} = I_{i,\sigma_i} \cdot r_{i,\sigma_i}$. Moreover, if the label of t_{ij} is $(g_{ij}, c_{ij}, \lambda_{ij})$, $\dot{Y}_{ij} = \lambda_{ij}(X_1, \ldots, X_k)$;
5. *activation condition* corresponding to t_{ij}, is the predicate $g_{ij} \wedge Y_{ij} \geq 1$, where g_{ij} is the guard predicate of the transition;
6. resets corresponding to t_{ij}, with $c_{ij} = \bigwedge_{k=1}^{h_{ij}} X_{i_k} = X_{i_k} + \delta_{ij}$,

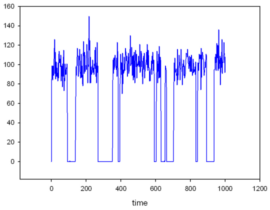
$$\left(\bigwedge_{k=1}^{h_{ij}} X'_{i_k} = X_{i_k} + \delta_{ij} \right) \wedge \left(\bigwedge_{t_{ij} \in T_i} Y'_{ij} = 0 \right).$$



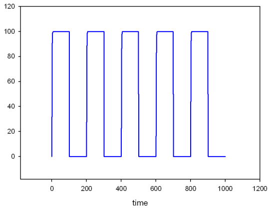
$[*; X = X + 1; k_+]$
$[X > 0; X = X - 1; k_- X]$
A
$[*; *; k_0]$
$[*; *; k_0]$
B
$[*; X = X + 1; k_-]$
$[X > 0; X = X - 1; k_+ X]$



$\dot{X} = k_+ - k_-$
$\dot{Y}_1 = k_0$
$\dot{Y}_2 = k_0$
$\dot{Y}_1 = 1$
$\dot{Y}_1' = 0$
$\dot{Y}_2' = 0$
$\dot{Y}_2 = 1$
$\dot{Y}_1' = 0$
$\dot{Y}_2' = 0$
$\dot{X} = k_- - k_+$
$\dot{Y}_1 = k_0$
$\dot{Y}_2 = k_0$



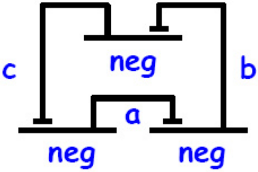
160
140
120
100
80
60
40
20
0
0
200
400
600
800
1000
1200
time



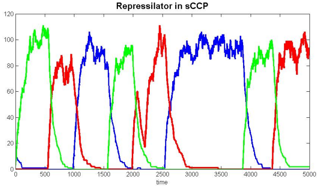
120
100
80
60
40
20
0
0
200
400
600
800
1000
1200
time

Hybrid Repressilator

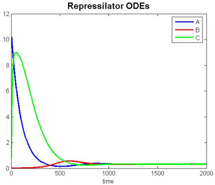
Repressilator with gene gates



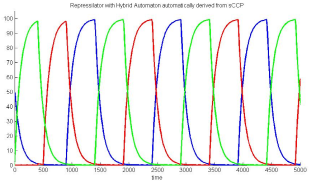
Repressilator in sCCP



Repressilator: ODE from sCCP



Hybrid Repressilator from sCCP



Conclusions

- HS for: biochemical reactions, genetic networks, etc.
- SPA to ODE: problems (the stochastic component *averaged away*).
- Localize ODE's and maintain a discrete portion of the network: Hybrid Systems (with the *right* control variables).

Future

- Define a *lattice* of HSs.
- Formalize the behavioral properties to guide/determine the level of discreteness to maintain.