Process Rate Semantics
Representing Biochemical Systems as Collectives of Interacting Automata

Luca Cardelli
Microsoft Research
Computational and Systems Biology Course
CoSBI, Trento, 2008-03-13

http://LucaCardelli.name
Aims

- **Connections between modeling approaches**
  - Connecting the *discrete/concurrent/stochastic/molecular* approach
  - to the *continuous/sequential/deterministic/population* approach

- **Connecting syntax with semantics**
  - Syntax = model (equations/programs/diagrams/blobs etc.)
  - Semantics = state space (generated by the syntax)
  - N.B. model ≠ state space !!
    - The same model can be interpreted in different state spaces
    - Different models can have the same state space
    - Different models of the same state space can support different analysis

- **Ultimately, connections between analysis techniques**
  - We need (and sometimes have) good semantic techniques to analyze state spaces (e.g. calculus, but also model-checking)
  - But we need equally good syntactic techniques to structure complex models (e.g. compositionality) and analyze them (e.g. process algebra)
Motivation: Cells Compute

- **No survival without computation!**
  - Finding food
  - Avoiding predators

- **How do they compute?**
  - Unusual computational paradigms.
  - Proteins: do they work like electronic circuits?
  - Genes: what kind of software is that?

- **Signaling networks**
  - Clearly “information processing”
  - They are “just chemistry”: molecule interactions
  - But what are their principles and algorithms?

- **Complex, higher-order interactions**
  - MAPKKK = MAP Kinase Kinase Kinase: that which operates on that which operates on that which operates on protein.

- **General models of biological computation**
  - What are the appropriate ones?

---

(Macro)Molecules as Interacting Automata

- Concurrent
  (math is based on processes, not functions)
- Asynchronous
  (no global clock)
- Stochastic
  (or nondeterministic)
- Stateful
  (e.g. phosphorylation state)
- Discrete
  (transitions between states)
- Interacting
  (an “interaction” can be pretty much anything you want that changes molecular state)

- Based on work on process algebra and biological modeling; see references in related papers.
Stochastic Automata Collectives

• “Collective“:
  - A large set of interacting finite state automata:
    • Not quite language automata ("large set")
    • Not quite cellular automata ("interacting" but not on a grid)
    • Not quite process algebra ("collective behavior")
    • Cf. multi-agent systems and swarm intelligence

• “Stochastic“:
  - Interactions have rates
    • Not quite discrete (hundreds or thousands of components)
    • Not quite continuous (non-trivial stochastic effects)
    • Not quite hybrid (no “switching” between regimes)

• Very much like biochemistry
  - Which is a large set of stochastically interacting molecules/proteins
  - Are proteins finite state and subject to automata-like transitions?
    • Let's say they are, at least because:
      • Much of the knowledge being accumulated in Systems Biology
        is described as state transition diagrams [Kitano].
Towards Systems Biology
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
<th>H</th>
<th>I</th>
<th>J</th>
<th>K</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Roche Applied Sciences Biochemical Pathways Wall Chart

http://www.expasy.ch/cgi-bin/show_thumbnails.pl
Interacting Automata

A_1 is a state

a is a channel i.e. a named interaction interface (e.g. a surface patch)

?,! indicate any complementarity of interaction (e.g. charge)

?a, !a indicate complementary actions,

@r, @s are rates
Interacting Automata

A_1 is a state

a is a channel i.e. a named interaction interface (e.g. a surface patch)

?!, indicate any complementarity of interaction (e.g. charge)

?a, !a indicate complementary actions, joined by an interaction arrow →

@r, @s are rates

Kinetic laws:

Two complementary actions may result in an interaction.
Kinetic laws:

Two complementary actions may result in an interaction.

A decay may happen spontaneously.

A\(_1\) is a state

\(a\) is a channel i.e. a named interaction interface (e.g. a surface patch)

\(?a, !a\) indicate any complementarity of interaction (e.g. charge)

\@r, @s\ are rates
Interacting Automata

The equivalent process algebra model

new $a@r_1$
new $b@r_2$
new $c@r_3$

$A_1 = ?a; A_2$
$A_2 = !c; A_3$
$A_3 = \tau@\lambda_5; A_1$

$B_1 = \tau@\lambda_2; B_2 + !a; B_3$
$B_2 = \tau@\lambda_1; B_1$
$B_3 = ?b; B_2$

$C_1 = !b; C_2 + ?c; C_3$
$C_2 = \tau@\lambda_3; C_1$
$C_3 = \tau@\lambda_4; C_2$

$A_1 \parallel B_1 \parallel C_1$

The system and initial state

Interactions have rates. Actions DO NOT have rates.
Suppose this is the next interaction (stochastically chosen)

One lonely automaton cannot interact
Interactions in a Population
Interactions in a Population

All-A stable population
Suppose this is the next interaction
Interactions in a Population (2)

All-B stable population

Nondeterministic population behavior ("multistability")
**CTMC Semantics**

CTMC (homogeneous) Continuous Time Markov Chain
- directed graph with no self loops
- nodes are system states
- arcs have transition rates

Probability of holding in state A:
\[ \Pr(H_A > t) = e^{-rt} \]

in general, \( \Pr(H_A > t) = e^{-Rt} \) where R is the sum of all the exit rates from A
Chemistry vs. Automata

A process algebra (chemistry)

\[ r: \ A + B \xrightarrow{k_1} C + D \]
\[ s: \ C + D \xrightarrow{k_2} A + B \]

A different process algebra (automata)

Does A become C or D?

A Petri-Net-like representation. Precise and dynamic, but not modular, scalable, or maintainable.

A compositional graphical representation (precise, dynamic and modular) and the corresponding calculus.

\[ A = \!r_{k_1} : C \]
\[ C = \?s_{k_2} : A \]
\[ B = \?r_{k_1} : D \]
\[ D = \!s_{k_2} : B \]

A reaction oriented 1 line per reaction

Interaction oriented 1 line per component
Groupies and Celebrities

Groupie
(wants to be like somebody different)

Celebrity
(does not want to be like somebody else)

| A stochastic collective of groupies: |
| A stochastic collective of celebrities: |

Stable because as soon as a A finds itself in the majority, it is more likely to find somebody in the same state, and hence change, so the majority is weakened.

Unstable because within an A majority, an A has difficulty finding a B to emulate, but the few B's have plenty of A's to emulate, so the majority may switch to B. Leads to deadlock when everybody is in the same state and there is nobody different to emulate.
Both Together

A way to break the deadlocks: Groupies with just a few Celebrities

Many Groupies

A few Celebrities

A tiny bit of “noise” can make a huge difference

Luca Cardelli
2008-03-13
Hysteric Groupies

We can get more regular behavior from groupies if they "need more convincing", or "hysteresis" (history-dependence), to switch states.

Regularity can arise not far from chaos

(With doping to break deadlocks)

N.B.: It will not oscillate without doping (noise)

"regular" oscillation
Semantics of Collective Behavior
The Two Semantic Sides of Chemistry

These diagrams commute via appropriate maps.

L. Cardelli: “On Process Rate Semantics” (TCS)
L. Cardelli: “A Process Algebra Master Equation” (QEST'07)
Quantitative Process Semantics

Continuous-state Semantics (Mass Action Kinetics)

\[ \frac{d[X]}{dt} = \sum_{Y \in E} \text{Accr}_E(Y, X) \cdot [Y] - \text{Depl}_E(X) \cdot [X] \quad \text{for all } X \in E \]

Discrete-state Semantics (Chemical Master Equation)

\[ \frac{\partial p_r(p,t)}{\partial t} = \sum_{v \in \mathcal{I}} a(p-v) \cdot p_r(p-v,t) - a(p) \cdot p_r(p,t) \quad \text{for all } p \in \text{States}(E) \]

Process Rate Equation

Nondeterministic Semantics

Defined over the syntax of processes

Accretion
Depletion

Interactions
Propensity
Stochastic Processes & Discrete Chemistry
Chemical Reactions

\[ A \rightarrow^{r} B_1 + \ldots + B_n \quad (n \geq 0) \]

Unary Reaction \( d[A]/dt = -r[A] \)

Exponential Decay

\[ A_1 + A_2 \rightarrow^{r} B_1 + \ldots + B_n \quad (n \geq 0) \]

Hetero Reaction \( d[A_i]/dt = -r[A_1][A_2] \)

Mass Action Law

\[ A + A \rightarrow^{r} B_1 + \ldots + B_n \quad (n \geq 0) \]

Homeo Reaction \( d[A]/dt = -2r[A]^2 \)

Mass Action Law

(assuming \( A \neq B \neq A \) for all \( i,j \))

No other reactions!

Trimolecular reactions:

\[ A + B + C \rightarrow^{r} D \]

the measured “\( r \)” is an (imperfect) aggregate of e.g.:

\[ A + B \leftrightarrow AB \]

\[ AB + C \rightarrow D \]

Enzymatic reactions:

\[ S \rightleftharpoons E \rightleftharpoons P \]

the “\( r \)” is given by Michaelis-Menten (approximated steady-state) laws:

\[ E + S \leftrightarrow ES \]

\[ ES \rightarrow P + E \]

Reactions have rates. Molecules do not have rates.
**Chemical Ground Form (CGF)**

\[
\begin{align*}
E &::= 0 : X=M, E \\
M &::= 0 : \pi;P \oplus M \\
P &::= 0 : X | P \\
\pi &::= \tau(r) : ?a(r) : !a(r) \\
CGF &::= E,P
\end{align*}
\]

- **Reagents**
- **Molecules**
- **Solutions**
- **Actions (delay, input, output)**

Reagents plus Initial Conditions

⊕ is stochastic choice (vs. + for chemical reactions)
0 is the null solution (P|0 = 0|P = P)
and null molecule (M⊕0 = 0⊕M = M)
Each X in E is a distinct species
Each name a is assigned a fixed rate r: a\(_{(r)}\)

A stochastic subset of CCS
(no values, no restriction)

Interacting Automata + dynamic forking

(To translate chemistry to processes we need a bit more than interacting automata: we may have “+” on the right of →, that is we may need “|” after π.)

**Ex: Interacting Automata**

(= finite-control CGFs: they use “|” only in initial conditions):

\[
\text{Automaton in state A} \\
\text{Automaton in state B} \\
\text{Initial conditions: 2A and 2B}
\]

\[
\begin{align*}
A &= !a;A \oplus ?b;B \\
B &= !b;B \oplus ?a;A \\
A|A|B|B
\end{align*}
\]
## From Reagents to Reactions (by example)

<table>
<thead>
<tr>
<th>Interacting Automata</th>
<th>Discrete Chemistry</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial states</td>
<td>initial quantities</td>
</tr>
<tr>
<td>$A</td>
<td>A</td>
</tr>
<tr>
<td>$A \xrightarrow{r} A'$</td>
<td>$A \rightarrow^r A'$</td>
</tr>
<tr>
<td>$A \xrightarrow{?a} A'$ $B \xleftarrow{!a} B'$</td>
<td>$A+B \rightarrow^r A'+B'$</td>
</tr>
<tr>
<td>$A \xrightarrow{?a} A'$ $A' \xrightarrow{!a} A''$</td>
<td>$A+A \rightarrow^{2r} A'+A''$</td>
</tr>
</tbody>
</table>

---

*Luca Cardelli*  
2008-03-13
From Reactions to Reagents (by example)

1: Fill the matrix by columns:

Degradation reaction $v_i: X \rightarrow_k P_i$
add $\tau_iP_i$ to $<X,v_i>$.

Hetero reaction $v_i: X+Y \rightarrow_k P_i$
add $?;P_i$ to $<X,v_i>$ and $!;0$ to $<Y,v_i>$

Homeo reaction $v_i: X+X \rightarrow_k P_i$
add $?;P_i$ and $!;0$ to $<X,v_i>$

2: Read the result by rows:

$A = ?v_{1(k1)};(C|C) \oplus ?v_{2(k2)};D$
$B = !v_{1(k1)};0$
$C = !v_{2(k2)};0 \oplus \tau_{k3};(E|F)$
$D = 0$
$E = 0$
$F = ?v_{4(k4/2)};B \oplus !v_{4(k4/2)};0$
We need a semantics of automata that identifies automata that have the "same chemistry". No process algebra equivalence is like this!

Entangled automata lead to more compact models than in chemistry. Detangled automata are in simple correspondence with chemistry.
Could chemistry itself be that semantics?
No: different sets of reactions can have the same behavior!

Different reactions, but they induce the same ODEs
Discrete-State Semantics
Discrete Semantics of Reactions

Syntax:

\[ A + B \rightarrow^r A + A \]
\[ A + B \rightarrow^r B + B \]
\[ A + B + B \]

Semantics:

\[ \{3B\} \]
\[ \{2A, 1B\} \]
\[ 2r_a \]
\[ 2r_b \]
\[ \{3A\} \]

CTMC
Discrete Semantics of Reagents

Syntax:

Semantics:

CTMC
Def: \( \equiv \) is equivalent CTMC's (isomorphic graphs with same rates).

Thm: \( E \equiv Ch(E) \)

Thm: \( C \equiv Pi(C) \)

For each \( E \) there is an \( E' \equiv E \) that is detangled (\( E' = Pi(Ch(E)) \)).

For each \( E \) in automata form there is an \( E' \equiv E \) that is detangled and in automata form (\( E' = Detangle(E) \)).
This is enough to establish that the process algebra is really faithful to the chemistry.

But CTMC are not the “ultimate semantics” because there are still questions of when two different CTMCs are actually equivalent (e.g. “lumping”).

The “ultimate semantics” of chemistry is the Chemical Master Equation (derivable from the Chapman-Kolmogorov equation of the CTMC).
Exercise: Making Lines

Or: build me a population like this:
Second-order and Zero-order Regime

Second-Order Regime
\[ [S]^* = -r[E][S] \]

directive sample 1000.0
directive plot \( S() ; P() ; E() \)

new \( a@1.0:\text{chan}() \)

let \( E() = !a; E() \)
and \( S() = ?a; P() \)
and \( P() = () \)

run (1 of \( E() \) | 1000 of \( S() \))

Zero-Order Regime
\[ [S]^* \approx -1 \quad (\text{by assuming } [ES]^* = 0) \]

directive sample 1000.0
directive plot \( S() ; P() ; E() \)

new \( a@1.0:\text{chan}() \)

let \( E() = !a; \text{delay}@1.0; E() \)
and \( S() = ?a; P() \)
and \( P() = () \)

run (1 of \( E() \) | 1000 of \( S() \))
**Cascades**

**Second-Order Regime cascade:** a signal amplifier (MAPK)
\[ a_{Hi} > 0 \Rightarrow c_{Hi} = \text{max} \]

**Zero-Order Regime cascade:** a signal divider!
\[ a_{Hi} = \text{max} \Rightarrow c_{Hi} = \frac{1}{3} \text{max} \]
Ultrasensitivity

Zero-Order Regime
A small E-F inbalance causes a much larger S-P switch.

Second-Order Regime
Continuous-State Semantics (short version)
## The Gillespie(?) Conversion

<table>
<thead>
<tr>
<th>Discrete Chemistry</th>
<th>Continuous Chemistry</th>
<th>$\gamma = N_A V : M^{-1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>initial quantities</td>
<td>initial concentrations</td>
<td>$[A]_0 \quad \text{with} \quad [A]_0 = #A_0/\gamma$</td>
</tr>
<tr>
<td>$A \rightarrow r A'$</td>
<td>$A \rightarrow^k A'$</td>
<td>$k = r : s^{-1}$</td>
</tr>
<tr>
<td>$A+B \rightarrow r A'+B'$</td>
<td>$A+B \rightarrow^k A'+B'$</td>
<td>$k = r\gamma : M^{-1}s^{-1}$</td>
</tr>
<tr>
<td>$A+A \rightarrow r A'+A''$</td>
<td>$A+A \rightarrow^k A'+A''$</td>
<td>$k = r\gamma/2 : M^{-1}s^{-1}$</td>
</tr>
</tbody>
</table>

$V = \text{interaction volume}$

$N_A = \text{Avogadro's number}$

Think $\gamma = 1$

i.e. $V = 1/N_A$

$M = \text{mol} \cdot L^{-1}$

molarity (concentration)
From Reactions to ODEs

$$v_1: A + B \rightarrow k_1 C + C$$
$$v_2: A + C \rightarrow k_2 D$$
$$v_3: C \rightarrow k_3 E + F$$
$$v_4: F + F \rightarrow k_4 B$$

Write the coefficients by columns

<table>
<thead>
<tr>
<th>Reaction</th>
<th>Stoichiometric Matrix</th>
</tr>
</thead>
</table>

Quantity changes

Stoichiometric Matrix

Rate laws

$$d[X]/dt = N \cdot l$$

$$d[A]/dt = -l_1 - l_2$$
$$d[B]/dt = -l_1 + l_4$$
$$d[C]/dt = 2l_1 - l_2 - l_3$$
$$d[D]/dt = l_2$$
$$d[E]/dt = l_3$$
$$d[F]/dt = l_3 - 2l_4$$

Set a rate law for each reaction (Degradation/Hetero/Homeo)

E.g. $$d[A]/dt = -k_1[A][B] - k_2[A][C]$$

Read the concentration changes from the rows

Continuous Chemistry

Discrete Chemistry

Process Algebra

ODE

CTMC

ODE

CTMC
"The change in process concentration (!!) for X at time t is: the sum over all possible (kinds of) processes Y of: the concentration at time t of Y times the accretion from Y to X minus the concentration at time t of X times the depletion of X to some other Y"

\[ \frac{d[X]}{dt} = (\sum (Y \in E) \text{Accr}_E(Y, X) \cdot [Y]) - \text{Depl}_E(X) \cdot [X] \]

for all \( X \in E \)

\[ \text{Depl}_E(X) = \sum (i: \text{E}.X.i=\tau(r);P) r + \sum (i: \text{E}.X.i=?a(r);P) r \gamma \cdot \text{OutsOn}_E(a) + \sum (i: \text{E}.X.i=!a(r);P) r \gamma \cdot \text{InsOn}_E(a) \]

\[ \text{Accr}_E(Y, X) = \sum (i: \text{E}.Y.i=\tau(r);P) \#X(P) \cdot r + \sum (i: \text{E}.Y.i=?a(r);P) \#X(P) \cdot r \gamma \cdot \text{OutsOn}_E(a) + \sum (i: \text{E}.Y.i=!a(r);P) \#X(P) \cdot r \gamma \cdot \text{InsOn}_E(a) \]

\[ \text{InsOn}_E(a) = \sum (Y \in E) \#(Y.i | E.Y.i=?a(r);P) \cdot [Y] \]

\[ \text{OutsOn}_E(a) = \sum (Y \in E) \#(Y.i | E.Y.i=!a(r);P) \cdot [Y] \]
**Continuous State Equivalence**

- **Def:** $\approx$ is equivalence of polynomials over the field of reals.

- **Thm:** $E \approx \text{Cont}(\text{Ch}(E))$

- **Thm:** $\text{Cont}(C) \approx \text{Pi}(C)$

- For each $E$ there is an $E' \approx E$ that is detangled ($E' = \text{Pi}(\text{Ch}(E))$)

- For each $E$ in automata form there is an $E' \approx E$ that is detangled and in automata form ($E' = \text{Detangle}(E)$).
Exercise: Making Waves

Or: build me a population like this:
Nonlinear Transition (NLT)

\[ A = \mathcal{C}(s); B \]

\[ B = \mathcal{C}(s); B \]

\[ A + B \rightarrow^s B + B \]

\[ [A]^* = -s[A][B] \]

\[ [B]^* = s[A][B] \]

N.B.: needs at least 1 B to "get started".

Luca Cardelli
2008-03-13

48
Two NLTs: Bell Shape

![Diagram of Bell Shape]

\[ [B]^* = [B][A][C] \]

\[ A = ?b_{(1)}; B \]
\[ B = !b_{(1)}; B \oplus ?c_{(1)}; C \]
\[ C = !c_{(1)}; C \]

\[ A + B \rightarrow^{1} B + B \]
\[ B + C \rightarrow^{1} C + C \]

\[ [A]^* = -[A][B] \]
\[ [B]^* = [A][B] - [B][C] \]
\[ [C]^* = [B][C] \]
NLT in a Cycle: Oscillator (unstable)

\[ A = \text{!}a(s); A \oplus \text{?}b(s); B \]
\[ B = \text{!}b(s); B \oplus \text{?}c(s); C \]
\[ C = \text{!}c(s); C \oplus \text{?}a(s); A \]

A+B →s B+B
B+C →s C+C
C+A →s A+A

\[ [A]^* = -s[A][B]+s[C][A] \]
\[ [B]^* = -s[B][C]+s[A][B] \]
\[ [C]^* = -s[C][A]+s[B][C] \]
Oscillator (stable)

```
oscillator stable

directive sample 0.1 1000

directive plot A1(); A2(); A3()

val r=1.0
val s=1.0

new a1@s:chan
new a2@s:chan
new a3@s:chan

let A1() = do !a1;A1() or delay@r;A2() or ?a2;

A2() = do !a2;A2() or delay@r;A3() or ?a3;

A3() = do !a3;A3() or delay@r;A1() or ?a1;

run 1000 of A1()
```

N.B. this does not deadlock!

\[ A = !a(s); A \oplus \tau_r; B \oplus ?b(s); A' \]
\[ A' = ?b(s); B \]
\[ B = !b(s); B \oplus \tau_r; C \oplus ?c(s); B' \]
\[ B' = ?c(s); C \]
\[ C = !c(s); C \oplus \tau_r; A \oplus ?a(s); C' \]
\[ C' = ?a(s); A \]

Sustained Deterministic Oscillation

Robust Stochastic Oscillation

\[ [A]^* = -r[A]-s[A][B]+r[C]+s[C'][A] \]
\[ [C]^* = -r[C]-s[C][A]+r[B]+s[B'][C] \]
\[ [A']^* = -s[A'][B] + s[A][B] \]
\[ [B']^* = -s[B'][C] + s[B][C] \]
\[ [C']^* = -s[C'][A] + s[C][A] \]
NLTs in Series: Soliton Propagation

!a_1
A_1

?a_2

A_0

?a_n

A_n

**LaTeX** code:

```latex
\text{directive sample 0.1 1000}
\text{directive plot A1(); A2(); A3(); A4(); A5(); A6(); A7(); A8(); A9(); A10(); A11(); A12(); A13())
\text{val r=1.0 val s=1.0}
new a2@s:chan new a3@s:chan new a4@s:chan new a5@s:chan
new a6@s:chan new a7@s:chan new a8@s:chan
new a9@s:chan new a10@s:chan new a11@s:chan
new a12@s:chan new a13@s:chan
let A1() = do delay@r;A2() or delay@r;A3() or ?a1;A1()
and A2() = do delay@r;A1() or delay@r;A3() or ?a2;A2()
and A3() = do delay@r;A1() or delay@r;A2() or ?a3;A3()
and A4() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or ?a4;A4()
and A5() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or ?a5;A5()
and A6() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or ?a6;A6()
and A7() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or delay@r;A6() or ?a7;A7()
and A8() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or delay@r;A6() or delay@r;A7() or ?a8;A8()
and A9() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or delay@r;A6() or delay@r;A7() or delay@r;A8() or ?a9;A9()
and A10() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or delay@r;A6() or delay@r;A7() or delay@r;A8() or delay@r;A9() or ?a10;A10()
and A11() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or delay@r;A6() or delay@r;A7() or delay@r;A8() or delay@r;A9() or delay@r;A10() or ?a11;A11()
and A12() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or delay@r;A6() or delay@r;A7() or delay@r;A8() or delay@r;A9() or delay@r;A10() or delay@r;A11() or ?a12;A12()
and A13() = do delay@r;A1() or delay@r;A2() or delay@r;A3() or delay@r;A4() or delay@r;A5() or delay@r;A6() or delay@r;A7() or delay@r;A8() or delay@r;A9() or delay@r;A10() or delay@r;A11() or delay@r;A12() or ?a13;A13()
run 10000 of A1()
```
GMA \neq CME

Semantics #1
Syntax
Semantics #2
A + A $\rightarrow^{2r} A$  $=$?  $A + A \rightarrow^{r} 0$

1*reaction rate $r\gamma$ because 1*A is lost in reaction.

2*reaction rate $r\gamma/2$ because 2*A are lost in reaction.

Law of Mass Action

Gillespie conversion

CTMC
... as Automata

\[ \frac{d[A]}{dt} = -2r \gamma A^2 \]

\[ k = \frac{r \gamma}{2} \]

\[ A + A \rightarrow 2k A \]

\[ [A]_0 = \frac{2}{\gamma} \]

\[ A = ?a_{(2r)}; 0 \oplus !a_{(2r)}; A \]

\[ A|A \]

\[ \frac{d[A]}{dt} = -4k[A]^2 \]

\[ k = \frac{r \gamma}{2} \]

\[ A + A \rightarrow 2r 0 \]

\[ [A]_0 = \frac{2}{\gamma} \]

\[ A = ?a_{(r)}; 0 \oplus !a_{(r)}; A \]

\[ A|A \]

Luca Cardelli

2008-03-13
Continuous vs. Discrete Groupies

(all with doping)

Matlab

SPiM

2000xA, 0xB, 1xA_d, 1xB_d, r = 1.0
And Yet It Moves


The Repressilator

Parametric representation

Neg(a, b) = ?a; Inh(a, b) ⊕ τ(ε); (Tr(b) | Neg(a, b))
Inh(a, b) = τ(η); Neg(a, b)
Tr(b) = !b; Tr(b) ⊕ τ(γ); 0
Neg(x(r), y(r)) | Neg(y(r), z(r)) | Neg(z(r), x(r))

Neg/x,y → ε Tr/y + Neg/x,y
Neg/y,z → ε Tr/z + Neg/y,z
Neg/z,x → ε Tr/x + Neg/z,x
Tr/x + Neg/x,y → ε Tr/x + Inh/x,y
Tr/y + Neg/y,z → ε Tr/y + Inh/y,z
Tr/z + Neg/z,x → ε Tr/z + Inh/z,x
Inh/x,y → ε Neg/x,y
Inh/y,z → ε Neg/y,z
Inh/z,x → ε Neg/z,x
Tr/x → 0
Tr/y → ε
Tr/z → 0
Neg/x,y + Neg/y,z + Neg/z,x

simplifying (N is the quantity of each of the 3 gates)

d[Neg/x,y]/dt = ηN – (η + r[Tr/x]) [Neg/x,y]
d[Neg/y,z]/dt = ηN – (η + r[Tr/y]) [Neg/y,z]
d[Neg/z,x]/dt = ηN – (η + r[Tr/z]) [Neg/z,x]
d[Tr/x]/dt = ε[Neg/x,z] – γ[Tr/x]
d[Tr/y]/dt = ε[Neg/x,y] – γ[Tr/y]
d[Tr/z]/dt = ε[Neg/y,z] – γ[Tr/z]

Analytically not an oscillator!

A fine stochastic oscillator over a wide range of parameters.

Neg(a, b) = ?a; Inh(a, b)
Tr(b) = !b; Tr(b)
Neg(a, b) = τ(ε); (Tr(b) | Neg(a, b))

interval/step [0:10:20000]
N=1, r=1.0,
ε=0.1,
η=0.001,
γ=0.001

The Repressilator

Parametric representation

Neg(a, b) = ?a; Inh(a, b) ⊕ τ(ε); (Tr(b) | Neg(a, b))
Inh(a, b) = τ(η); Neg(a, b)
Tr(b) = !b; Tr(b) ⊕ τ(γ); 0
Neg(x(r), y(r)) | Neg(y(r), z(r)) | Neg(z(r), x(r))

Neg/x,y → ε Tr/y + Neg/x,y
Neg/y,z → ε Tr/z + Neg/y,z
Neg/z,x → ε Tr/x + Neg/z,x
Tr/x + Neg/x,y → ε Tr/x + Inh/x,y
Tr/y + Neg/y,z → ε Tr/y + Inh/y,z
Tr/z + Neg/z,x → ε Tr/z + Inh/z,x
Inh/x,y → ε Neg/x,y
Inh/y,z → ε Neg/y,z
Inh/z,x → ε Neg/z,x
Tr/x → 0
Tr/y → ε
Tr/z → 0
Neg/x,y + Neg/y,z + Neg/z,x

simplifying (N is the quantity of each of the 3 gates)

d[Neg/x,y]/dt = ηN – (η + r[Tr/x]) [Neg/x,y]
d[Neg/y,z]/dt = ηN – (η + r[Tr/y]) [Neg/y,z]
d[Neg/z,x]/dt = ηN – (η + r[Tr/z]) [Neg/z,x]
d[Tr/x]/dt = ε[Neg/x,z] – γ[Tr/x]
d[Tr/y]/dt = ε[Neg/x,y] – γ[Tr/y]
d[Tr/z]/dt = ε[Neg/y,z] – γ[Tr/z]
Model Compactness

\[ \text{ODE} \quad = \quad \text{ODE} \]

\[ \text{Continuous Chemistry} \quad \rightarrow \quad \text{Discrete Chemistry} \quad \rightarrow \quad \text{CTMC} \]

\[ \text{Process Algebra} \quad \rightarrow \quad \text{CTMC} \]
Entangled vs detangled

Detangle($E_3$) (closely related to $\Pi(Ch(E_3))$)
**n^2 Scaling Problems**

- $E_n$ has $2n$ variables (nodes) and $2n$ terms (arcs).
- $Ch(E_n)$ has $2n$ species and $n^2$ reactions.

- The stoichiometric matrix has size $2n \cdot n^2 = 2n^3$.
- The ODEs have $2n$ variables and $2n(n+n) = 4n^2$ terms (number of variables times number of accretions plus depletions when sums are distributed).

### Stoichiometric Matrix ($Ch(E_3)$)

<table>
<thead>
<tr>
<th></th>
<th>$a_{00}$</th>
<th>$a_{01}$</th>
<th>$a_{02}$</th>
<th>$a_{10}$</th>
<th>$a_{11}$</th>
<th>$a_{20}$</th>
<th>$a_{21}$</th>
<th>$a_{22}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$X_0$</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$X_1$</td>
<td>+1</td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$X_2$</td>
<td></td>
<td>+1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>$Y_0$</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>$Y_1$</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
<tr>
<td>$Y_2$</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>+1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

### ODEs ($E_3$)

$$
\begin{align*}
\frac{d[X_0]}{dt} &= -r[X_0][Y_0] - r[X_0][Y_1] - r[X_0][Y_2] + r[X_2][Y_0] + r[X_2][Y_1] + r[X_2][Y_2] \\
\frac{d[X_1]}{dt} &= -r[X_1][Y_0] - r[X_1][Y_1] - r[X_1][Y_2] + r[X_0][Y_0] + r[X_0][Y_1] + r[X_0][Y_2] \\
\frac{d[X_2]}{dt} &= -r[X_2][Y_0] - r[X_2][Y_1] - r[X_2][Y_2] + r[X_1][Y_0] + r[X_1][Y_1] + r[X_1][Y_2] \\
\frac{d[Y_0]}{dt} &= -r[X_0][Y_0] - r[X_1][Y_0] - r[X_2][Y_0] + r[X_0][Y_2] + r[X_1][Y_2] + r[X_2][Y_2] \\
\frac{d[Y_1]}{dt} &= -r[X_0][Y_1] - r[X_1][Y_1] - r[X_2][Y_1] + r[X_0][Y_0] + r[X_1][Y_0] + r[X_2][Y_0] \\
\frac{d[Y_2]}{dt} &= -r[X_0][Y_2] - r[X_1][Y_2] - r[X_2][Y_2] + r[X_0][Y_1] + r[X_1][Y_1] + r[X_2][Y_1]
\end{align*}
$$
Conclusions
Conclusions

- **Compositional models**
  - Accurate (at the “appropriate” abstraction level).
  - Manageable (so we can scale them up by composition).
  - Executable (stochastic simulation).

- **Analysis techniques**
  - Mathematical techniques: Markov theory, Chemical Master Equation, and Rate Equation
  - Computing techniques: Abstraction and Refinement, Model Checking, Causality Analysis.

- **Many lines of extensions**
  - Parametric processes for model factorization
  - Polyautomata for Bio-Chemistry: complexation and polymerization
  - Ultimately, rich process-algebra based modeling languages.

- **Quantitative techniques**
  - Important in the “real sciences”.