Artificial Biochemistry

Luca Cardelli
Microsoft Research

Understanding Adaptive Multi-Component Systems
Rostock 2006-11-10

http://LucaCardelli.name
Reverse-Engineer This!

Eukaryotic Cell
(10~100 trillion in human body)
Membranes everywhere

Reverse-Engineer This!

DNA inside

H. Lodish et al.
Molecular Cell Biology
fourth edition p.1
…and Model It

- Even if we understood it, how would we model it?
  - Millions of differential equations? Hmmm..

- And we will have to model it in order to understand it.

- What’s different about modeling these systems?
Stochastic Collectives
Stochastic 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 (“finite state” and “collective”)
    ● 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].
Even More State Transitions

http://www.expasy.ch/cgi-bin/show_thumbnails.pl
Communicating automata: a graphical FSA-like notation for “finite state restriction-free π-calculus processes”. Interacting automata do not even exchange values on communication.

The stochastic version has rates on communications, and delays.

“Finite state” means: no composition or restriction inside recursion.

Analyzable by standard Markovian techniques, by first computing the “product automaton” to obtain the underlying finite Markov transition system. [Buchholz]
Q: What kind of mass behavior can this produce?

(We need to understand that if we want to understand biochemical systems.)
Suppose this one is the next interaction

One lonely automaton can do no interactions
Interactions in a Population
Interactions in a Population
Suppose this one is the next interaction.
Interactions in a Population (2)
**Groupies and Celebrities**

**Celebrity**
(does not want to be like somebody else)

- directive sample 0.1 200
- directive plot A(); B()
- new a@1.0:chan()
- new b@1.0:chan()
- let A() = do !a; A() or ?a; B()
- and B() = do !b; B() or ?b; A()
- run 100 of (A() | B())

**Groupie**
(wants to be like somebody different)

- directive sample 0.1 200
- directive plot A(); B()
- new a@1.0:chan()
- new b@1.0:chan()
- let A() = do !a; A() or ?a; B()
- and B() = do !b; B() or ?b; A()
- run 100 of (A() | B())

A stochastic collective of celebrities:

A stochastic collective of groupies:

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.

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.
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

-directive sample 10.0 1000
-directive plot Ga(); Gb(); Ca(); Cb()

new a@1.0:chan()
new b@1.0:chan()

let Ca() = do !a; Ca() or ?a; Cb()
and Cb() = do !b; Cb() or ?b; Ca()

let Ga() = do !a; Ga() or ?b; Gb()
and Gb() = do !b; Gb() or ?a; Ga()

run 1 of (Ca() | Cb())
run 100 of (Ga() | Gb())

never deadlock

Ga vs. Gb
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

\[
\begin{align*}
\text{let } Ga() &= \text{do } !a; Ga() \text{ or } ?b; ?b; Gb() \\
\text{and } Gb() &= \text{do } !b; Gb() \text{ or } ?a; ?a; Ga() \\
\text{let } Da() &= !a; Da() \\
\text{and } Db() &= !b; Db() \\
\text{run } 100 \text{ of } (Ga() | Gb()) \\
\text{run } 1 \text{ of } (Da() | Db())
\end{align*}
\]
Hysteric 3-Way Groupies

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

directive sample 3.0 1000
directive plot A(); B(); C()

new a@1.0:chan()
new b@1.0:chan()
new c@1.0:chan()

let A() = do !a; A() or ?c; C()
and B() = do !b; B() or ?a; A()
and C() = do !c; C() or ?b; B()

let Da() = !a; Da()
and Db() = !b; Db()
and Dc() = !c; Dc()

run 100 of (A() | B() | C())
run 1 of (Da() | Db() | Dc())
Semantics of Collective Behavior
“Micromodels”: Continuous Time Markov Chains

- The underlying semantics of stochastic $\pi$-calculus (and stochastic interacting automata). Well established in many ways.
  - Automata with rates on transitions.

- “The” correct semantics for chemistry, executable.
  - Gillespie stochastic simulation algorithm

- Lots of advantages
  - Compositional, compact, mechanistic, etc.

- But do not give a good sense of “collective” properties.
  - Yes one can do simulation.
  - Yes one can do program analysis.
  - Yes one can do modelchecking.
  - But somewhat lacking in “analytical properties” and “predictive power”.

They always ask:
  - “Yes, but how does your automata model relate to the 75 ODE models in the literature?”

Going from processes/automata to ODEs directly:
  - *In principle*: just write down the Rate Equation: [Calder, Hillston]
    - Determine the set of all possible states $S$ of each process.
    - Determine the rates of the transitions between such states.
    - Let $[S]$ be the “number of processes in state $S$” as a function of time.
    - Define for each state $S$:
      $$[S] = \text{(rate of change of the number of processes in state } S)$$
      Cumulative rate of transitions from any state $S'$ to state $S$, times $[S']$,
      minus cumulative rate of transitions from $S$ to any state $S''$, times $[S]$.
  - Intuitive (rate = inflow minus outflow), but often clumsy to write down precisely.

But why go to the trouble?
  - If we first convert processes to chemical reactions, then we can convert to ODEs by standard means!
From Chemistry to ODEs
**Chemical Reactions**

\[
A \rightarrow^r B_1 + \ldots + B_n \quad \text{Degradation} \\
A_1 + A_2 \rightarrow^r B_1 + \ldots + B_n \quad \text{Asymmetric Collision} \\
A + A \rightarrow^r B_1 + \ldots + B_n \quad \text{Symmetric Collision}
\]

\[ [A]^* = -r[A] \quad \text{Exponential Decay} \]
\[ [A_i]^* = -r[A_1][A_2] \quad \text{Mass Action Law} \]
\[ [A]^* = -r[A][A]-1 \quad \text{Mass Action Law} \]

(assuming \( A \neq B_i \neq A_j \) 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:**

\[
S \rightarrow^E P
\]

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

\[
E + S \leftrightarrow ES \\
ES \rightarrow P + E
\]

---

Chapter IV: Chemical Kinetics

[David A. Reckhow, CEE 572 Course]

... reactions may be either elementary or non-elementary. Elementary reactions are those reactions that occur exactly as they are written, without any intermediate steps. These reactions almost always involve just one or two reactants. Non-elementary reactions involve a series of two or more elementary reactions. Many complex environmental reactions are non-elementary. In general, reactions with an overall reaction order greater than two, or reactions with some non-integer reaction order are non-elementary.

---

The Collision Theory of Reaction Rates

[www.chemguide.co.uk](http://www.chemguide.co.uk)

The chances of all this happening if your reaction needed a collision involving more than 2 particles are remote. All three (or more) particles would have to arrive at exactly the same point in space at the same time, with everything lined up exactly right, and having enough energy to react. That's not likely to happen very often!
From Reactions to ODEs

Write the coefficients by columns

\[ \begin{align*}
N & \quad v_1 & \quad v_2 & \quad v_3 & \quad v_4 \\
A & -1 & -1 & \quad \cdot & \quad \cdot \\
B & -1 & 1 & \quad \cdot & \quad \cdot \\
C & 2 & -1 & -1 & \quad \cdot \\
D & 1 & & & \quad \cdot \\
E & 1 & & & \quad \cdot \\
F & 1 & -2 & & \quad \cdot 
\end{align*} \]

Read the concentration changes from the rows

\[ [X] = N \cdot \text{I} \]

Stoichiometric Matrix

\[ \begin{align*}
\text{Species} & \quad A & \quad B & \quad C & \quad D & \quad E & \quad F \\
\text{Quantity changes} & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot \\
\text{Stoichiometric matrix} & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot \\
\text{Rate laws} & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot & \quad \cdot \\
\end{align*} \]

E.g. \[ [A] = -k_1[A][B] - k_2[A][C] \]

Set a rate law for each reaction (Degradation/Asymmetric/Symmetric)

\[ \begin{align*}
X: & \text{chemical species} \\
[-]: & \text{quantity of molecules} \\
l: & \text{rate laws} \\
k: & \text{kinetic parameters} \\
N: & \text{stoichiometric matrix} \\
\end{align*} \]

CAVEAT: A deterministic approximation of a stochastic system (i.e. possibly misleading)
From Processes to Chemistry
**Chemical Ground Form (CGF)**

\[ E ::= X_1=M_1, \ldots, X_n=M_n \]
\[ M ::= \pi_1;P_1 \oplus \ldots \oplus \pi_n;P_n \]
\[ P ::= X_1 | \ldots | X_n \]
\[ \pi ::= \tau_r ?n(r) !n(r) \]
\[ CGF ::= E,P \]

Definitions \((n \geq 0)\)

Molecules \((n \geq 0)\)

Solutions \((n \geq 0)\)

Interactions (delay, input, output)

Definitions with Initial Conditions

(To translate chemistry back to processes we need a bit more than simple automata: we may have \(\oplus\) on the right of \(\rightarrow\), that is we may need \(|\) after \(\pi\).

\(\oplus\) is stochastic choice (vs. + for chemical reactions)

0 is the null solution \((P|0 = 0|P = P)\)

and null molecule \((M \oplus 0 = 0 \oplus M = M)\) \((\tau_0;P = 0)\)

\(X_i\) are distinct in \(E\)

Each name \(n\) is assigned a fixed rate \(r: n_{(r)}\)

Ex: interacting automata

(\(\text{which are CGFs using } |\text{ only in initial conditions}\)): 

\(A = !a;A \oplus ?b;B \)

\(B = !b;B \oplus ?a;A \)

\(A|A|B|B\)

Automaton in state \(A\)

Automaton in state \(B\)

Initial conditions: \(2A\) and \(2B\)
Automata to Chemistry

\[ A \rightarrow^r A' \]

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

\[ A + A \rightarrow^{2r} A' + A'' \]
Process Rate Semantics
Same Chemistry

Same chemistry, hence equivalent automata

\[ B \rightarrow^s A \]
\[ A + B \rightarrow^r A + A \]
\[ A + A \rightarrow^{2r} A + B \]
Same ODEs

\[ \tau: B \rightarrow^t A \]
\[ a: A+B \rightarrow^r A+A \]
\[ b: A+A \rightarrow^{2r} A+B \]

Semantic Relationships

Processes $\approx$ Chemistry

Effective Rate

OpSem $\approx$ ODEs $\approx$ ODEs

Law of Mass Action

Rate Equation

Hermanns

CTMCs

Gillespie
Epidemics


Developing the Use of Process Algebra in the Derivation and Analysis of Mathematical Models of Infectious Disease

R. Norman and C. Shankland

Department of Computing Science and Mathematics, University of Stirling, UK.
{ces,ran}@cs.stir.ac.uk

Abstract. We introduce a series of descriptions of disease spread using the process algebra WSCCS and compare the derived mean field equations with the traditional ordinary differential equation model. Even the preliminary work presented here brings to light interesting theoretical questions about the “best” way to define the model.
ODE

\[ S = ?i(t);I \]
\[ I = !i(t);I \oplus ?i(t);I \oplus \tau_r;R \]
\[ R = ?i(t);R \]

\[ S + I \rightarrow^t I + I \]
\[ I + I \rightarrow^t I + I \]
\[ I \rightarrow^r R \]
\[ R + I \rightarrow^t R + I \]

\[
\begin{align*}
[S]^* &= -t[S][I] \\
[I]^* &= t[S][I] - r[I] \\
[R]^* &= r[I]
\end{align*}
\]

**Automata match the standard ODE model!**

(the Kermack-McKendrick, or SIR model)
Simplified Model

The simplified model of the infection process can be represented by the following equations:

\[ S = ?i(t); I \]
\[ I = !i(t); I \oplus \tau_r; R \]
\[ R = 0 \]

The transition rules are:

- \( S + I \rightarrow^+ I + I \)
- \( I \rightarrow^r R \)

The ODE system is:

\[ [S]^* = -t[S][I] \]
\[ [I]^* = +t[S][I] - r[I] \]
\[ [R]^* = r[I] \]

These ODEs lead to the same dynamics as the automata model, hence the two models are equivalent.

Not totally obvious that one could have simplified the automata model.

Useful directive sample and plot:

```
directive sample 500.0 1000
directive plot Recovered(); Susceptible(); Infected()
new infect @0.001:chan()
val recover = 0.03
let Recovered() = ()
and Susceptible() = ?infect; Infected()
and Infected() = do !infect; Infected()
or delay@recover; Recovered()
run (200 of Susceptible() | 2 of Infected())
```
Lotka-Volterra
Predator-Prey

```
directive sample 1.0 1000
directive plot Carnivor(); Herbivor()

val mortality = 100.0
val breeding = 300.0
val predation = 1.0
new cull @predation:chan()

let Herbivor() =
do delay@breeding; (Herbivor() | Herbivor()) or ?cull; ()
and Carnivor() =
do delay@mortality; () or !cull; (Carnivor() | Carnivor())

run 100 of Herbivor()
run 100 of Carnivor()
```

An unbounded state system!
ODE

\[ H = \tau_b : (H|H) \oplus ?c(p) : 0 \]
\[ C = \tau_m : 0 \oplus !c(p) : (C|C) \]

\[ H \rightarrow^b H + H \]
\[ C \rightarrow^m 0 \]
\[ H + C \rightarrow^p C + C \]

\([H]^* = b[H] - p[H][C] \]
\([C]^* = -m[C] + p[H][C] \]

**Lotka-Volterra Equations**

The Lotka-Volterra equations describe an ecological predator-prey (or parasite-host) model which assumes that, for a set of fixed positive constants \( A \) (the growth rate of prey), \( B \) (the rate at which predators destroy prey), \( C \) (the death rate of predators), and \( D \) (the rate at which predators increase by consuming prey), the following conditions hold:

1. A prey population \( x \) increases at a rate \( \frac{dx}{dt} = A x - B x y \) (proportional to the number of prey) but is simultaneously destroyed by predators at a rate \( \frac{dx}{dt} = -B x y d t \) (proportional to the product of the numbers of prey and predators).

2. A predator population \( y \) decreases at a rate \( \frac{dy}{dt} = -B x y d t \) (proportional to the number of predators), but increases at a rate \( \frac{dy}{dt} = D x y d t \) (again proportional to the product of the numbers of prey and predators).

This gives the coupled differential equations

\[ \frac{dx}{dt} = A x - B x y \quad (1) \]
\[ \frac{dy}{dt} = -C y + D x y \quad (2) \]

**Automata match the Lotka-Volterra model (with B=D)**
Laws by ODEs
Idle Delay Law by ODEs

\[ A = \tau_\lambda; A \oplus \tau_\mu; B = A = \tau_\mu; B \]

\[ A = \tau_\lambda; A \oplus \tau_\mu; B \]

\[ A \rightarrow^\lambda A \]

\[ A \rightarrow^\mu B \]

\[ [A]^* = -\mu[A] \]

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

\[ A = \tau_\mu; B \]

\[ A \rightarrow^\mu B \]

\[ [A]^* = -\mu[A] \]

\[ [B]^* = \mu[A] \]
Idle Interaction Law by ODEs

It may seem like \( A \) should decrease half as fast, but NO! Two ways to explain:

- State \( A \) is memoryless of any past idling.
- Activity on \( c \) is double

\[
A = ?c; B \\
C = !c; C
\]

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

- State \( A \) is memoryless of any past idling.
- Activity on \( c \) is double

\[
A = ?c; A \oplus ?c; B \\
C = !c; C
\]

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

\[ [A]^*_\cdot = -r[A][C] \\
[B]^*_\cdot = r[A][C] \\
[C]^*_\cdot = 0 \]
Asynchronous Interleaving

\[ \tau_{\lambda}; B | \tau_{\mu}; D = \tau_{\lambda}; (B | \tau_{\mu}; D) + \tau_{\mu}; (\tau_{\lambda}; B | D) \]

Amazingly, the B’s and the D’s from the two branches sum up to exponential distributions

Hermanns: Interactive Markov Chains. Sec 4.1.2
Asynchronous Interleaving Law by ODEs

\[ \tau_\lambda;B \mid \tau_\mu;D = \tau_\lambda;(B \mid \tau_\mu;D) + \tau_\mu;(\tau_\lambda;B \mid D) \]

Want to show that B and D on both sides have the “same behavior” (equal quantities of B and D produced at all times)

\[ A_1 = \tau_\lambda;B \]
\[ C_1 = \tau_\mu;D \]
\[ A_1 \mid C_1 \]
\[ A_1 \rightarrow^\lambda B \]
\[ C_1 \rightarrow^\mu D \]
\[ A_1 + C_1 \]
\[ [A_1]^* = -\lambda[A_1] \]
\[ [B]^* = \lambda[A_1] \]
\[ [C_1]^* = -\mu[C_1] \]
\[ [D]^* = \mu[C_1] \]

\[ Y = \tau_\lambda;(B \mid C_2) \oplus \tau_\mu;(A_2 \mid D) \]
\[ C_2 = \tau_\mu;D \]
\[ A_2 = \tau_\lambda;B \]
\[ Y \]
\[ Y \rightarrow^\lambda B + C_2 \]
\[ Y \rightarrow^\mu A_2 + D \]
\[ C_2 \rightarrow^\mu D \]
\[ A_2 \rightarrow^\lambda B \]
\[ [Y]^* = -\lambda[Y] - \mu[Y] \]
\[ [A_2]^* = \mu[Y] - \lambda[A_2] \]
\[ [B]^* = \lambda[Y] + \lambda[A_2] \]
\[ [C_2]^* = \lambda[Y] - \mu[C_2] \]
\[ [D]^* = \mu[Y] + \mu[C_2] \]

\[ [Y+A_2]^* = \lambda[Y+A_2] \]
\[ [B]^* = \lambda[Y+A_2] \]
\[ [Y+C_2]^* = -\mu[Y+C_2] \]
\[ [D]^* = \mu[Y+C_2] \]

\[ [Y+A_2]^* = [Y]^* + [A_2]^* \]
\[ = -\lambda[Y] - \mu[Y] + [Y]^* - \lambda[A_2] \]
\[ = -\lambda[Y] - \lambda[A_2] \]
\[ = -\lambda[Y+A_2] \]

\[ [Y+A_2] \text{ decays exponentially!} \]

\[ [B] \text{ and } [D] \text{ have equal time evolutions on the two sides provided that } [A_1]=[Y+A_2] \text{ and } [C_1]=[Y+C_2]. \]

This imposes the constraint, in particular, that \([A_1]_0=[Y+A_2]_0\) and \([C_1]_0=[Y+C_2]_0\) (at time zero).

The initial conditions of the right hand system specify that \([A_2]_0=[C_2]_0=0\) (since only Y is present).

Therefore, we obtain that \([A_1]_0=[C_1]_0=[Y]_0\).

So, for example, if we run a stochastic simulation of the left hand side with 1000*A1 and 1000*C1, we obtain the same curves for B and D than a stochastic simulation of the right hand side with 1000*Y.
Polymerization
new c@μ  new stop@1.0

A_{free} =
!c(\nu rht); A_{brht}(rht)) +
?c(lft); A_{blft}(lft)

A_{blft}(lft) =
!c(\nu rht); A_{bound}(lft,rht))

A_{brht}(rht) =
?c(lft); A_{bound}(lft,rht))

A_{bound}(lft,rht) = ?stop

Polymerization is iterated complexation.

Communicating Automata
Bound output \!c(\nu r) and input ?c(l)
on automata transitions
to model complexation.
Bidirectional Polymerization

Circular Polymer Lengths

Scanning and counting the size of the circular polymers (by a cheap trick).
Polymer formation is complete within 10t; then a different polymer is scanned every 100t.

100xA_free, initially.
The height of each rising step is the size of a separate circular polymer.
(Unbiased sample of nine consecutive runs.)
new c@μ

\[ A_{\text{free}} = \nonline \text{(!c(lft); } A_{\text{blft}(lft)) } + \text{ (?c(rht); } A_{\text{brht}(rht)} \nonline \] 

\[ A_{\text{blft}(lft)} = \nonline \text{(!lft; } A_{\text{free} } + \text{ (?c(rht); } A_{\text{bound}(lft, rht)} \nonline \] 

\[ A_{\text{brht}(rht)} = \nonline \text{(?rht; } A_{\text{free}} \nonline \] 

\[ A_{\text{bound}(lft, rht)} = \nonline \text{(!lft; } A_{\text{brht}(rht)} \nonline \] 

1000 monomers settle to ~100 polymers of size ~10
Conclusions
Conclusions

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

- **Interacting Automata**
  - Complex global behavior from simple components.
  - Bridging individual and collective behavior.
  - Connections to classical Markov theory, chemical Master Equation, and Rate Equation.

- **Mapping out “the whole system”**
  - A bit at a time, and simultaneously at different levels.
  - For prediction and prevention.
  - Through an “artificial biochemistry” (a scalable mathematical and computational modeling framework) to investigate “real biochemistry” on a large scale.