Artificial Biochemistry

Luca Cardelli

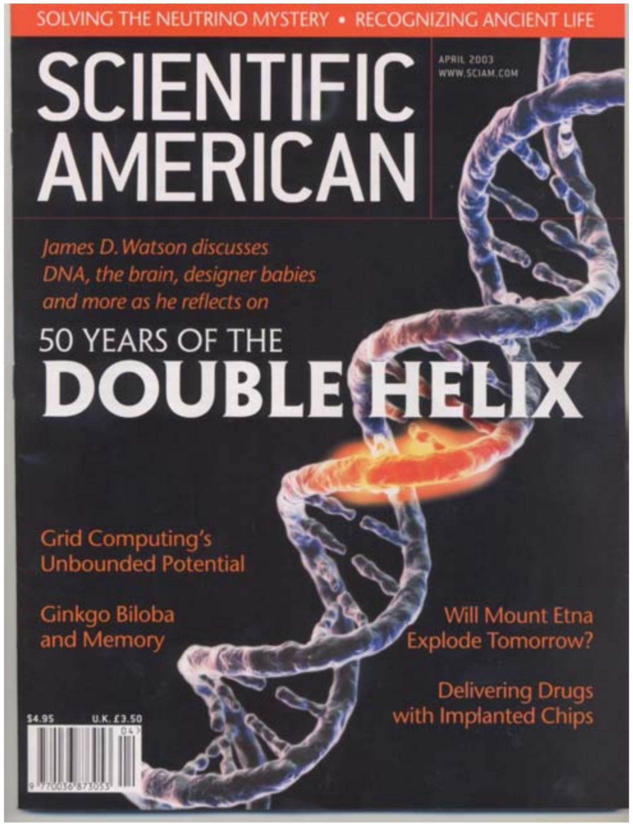
Microsoft Research

Distinguished Lecture Series
Computer and Information Science
University of Pennsylvania 2007-05-29

http://LucaCardelli.name

50 Years of Molecular Cell Biology

- The genome (human: 3 GBases = 750MB) is made of DNA
  - Stores digital information as sequences of 4 different nucleotides
  - Directs protein assembly through RNA and the Genetic Code
- Proteins (~1M coded from ~30K genes) are made of amino acids strings
  - Catalyze all biochemical reactions
  - Control metabolism (energy & materials)
  - Process signals, activate genes
- Bootstrapping still a mystery
  - DNA, RNA, proteins, membranes are today interdependent. Not clear who came first
  - Not understood, not essential for us



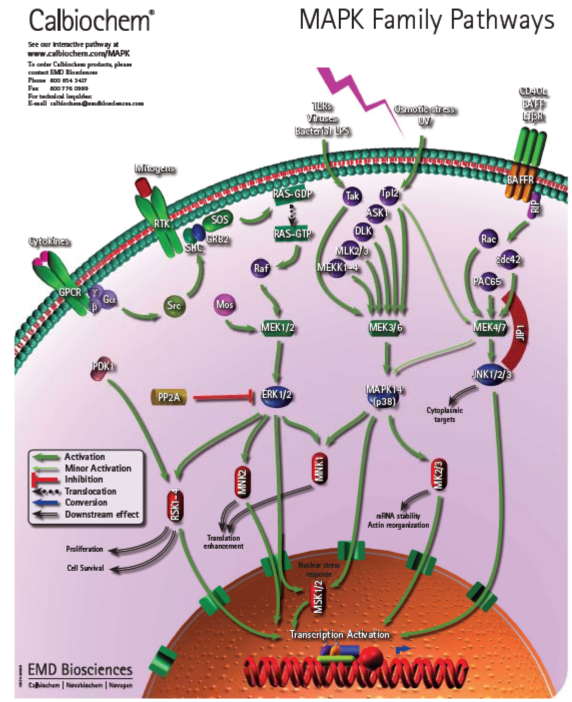
Cells Compute

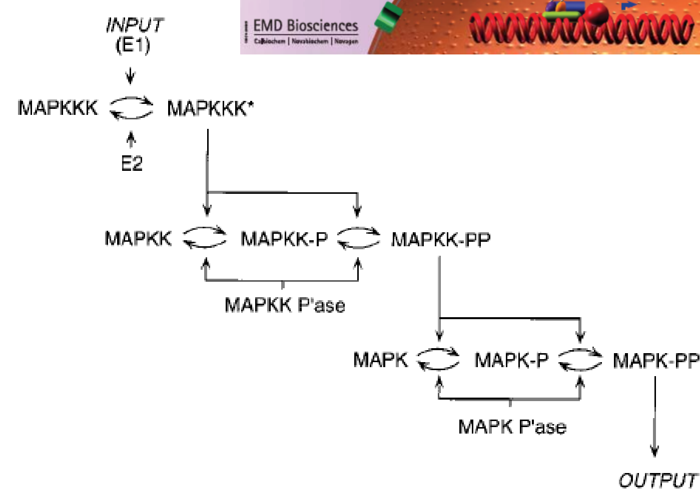
- If they don't, they die
  - Finding food (information processing)
  - Avoiding predators (information processing)

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

- Signaling networks
  - Clearly "information processing"
  - They are "just chemistry": molecule interactions
    - Much simpler than gene networks, neural networks, ants, and bees!
  - 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.





Ultrasensitivity in the mitogen-activated protein **cascade**, Chi-Ying F. Huang and James E. Ferrell, Jr., 1996, *Proc. Natl. Acad. Sci. USA*, 93, 10078-10083.

Stochastic Collectives

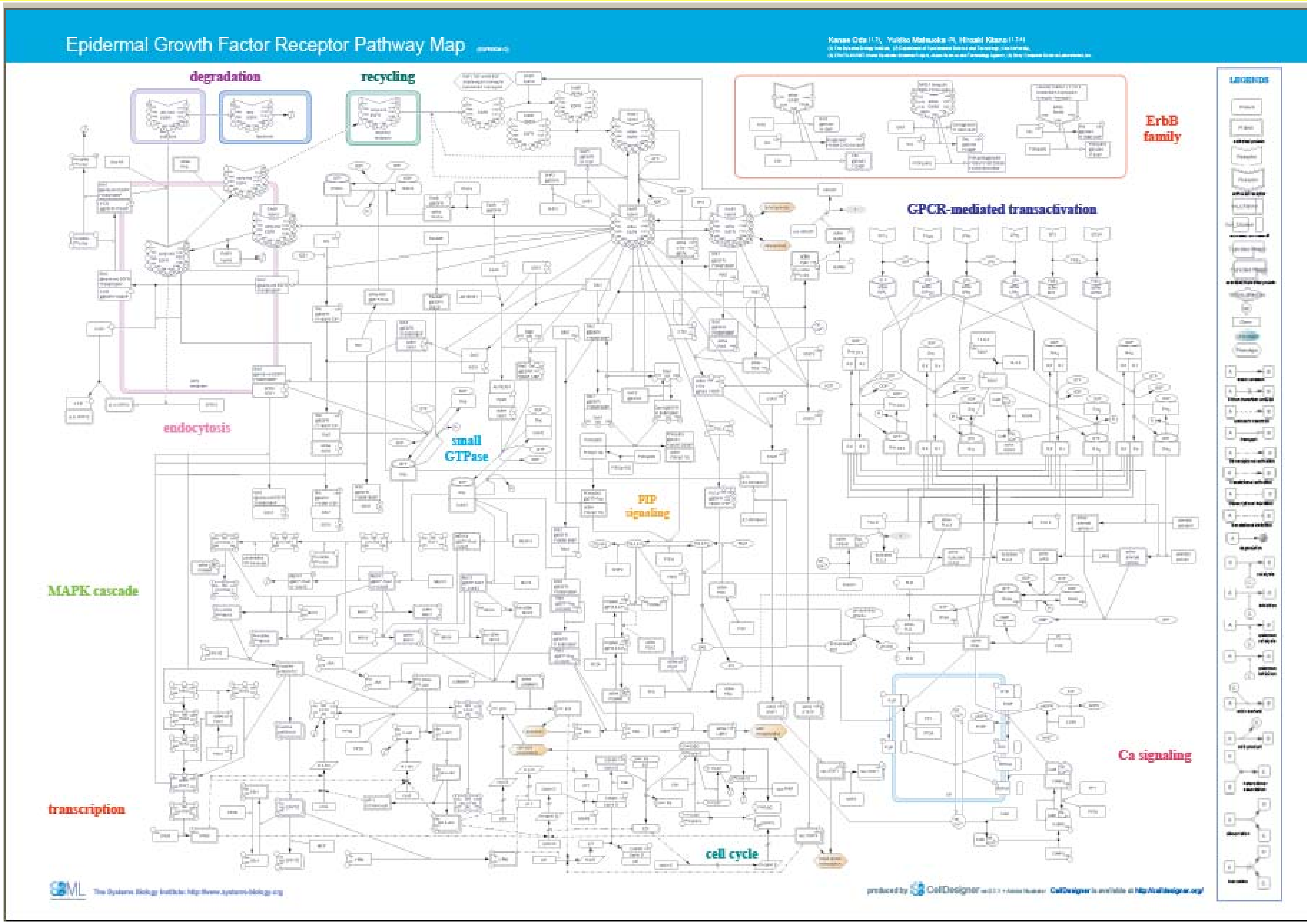
Stochastic Collectives

- "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 ("collective behavior")
    - Cf. multi-agent systems and swarm intelligence

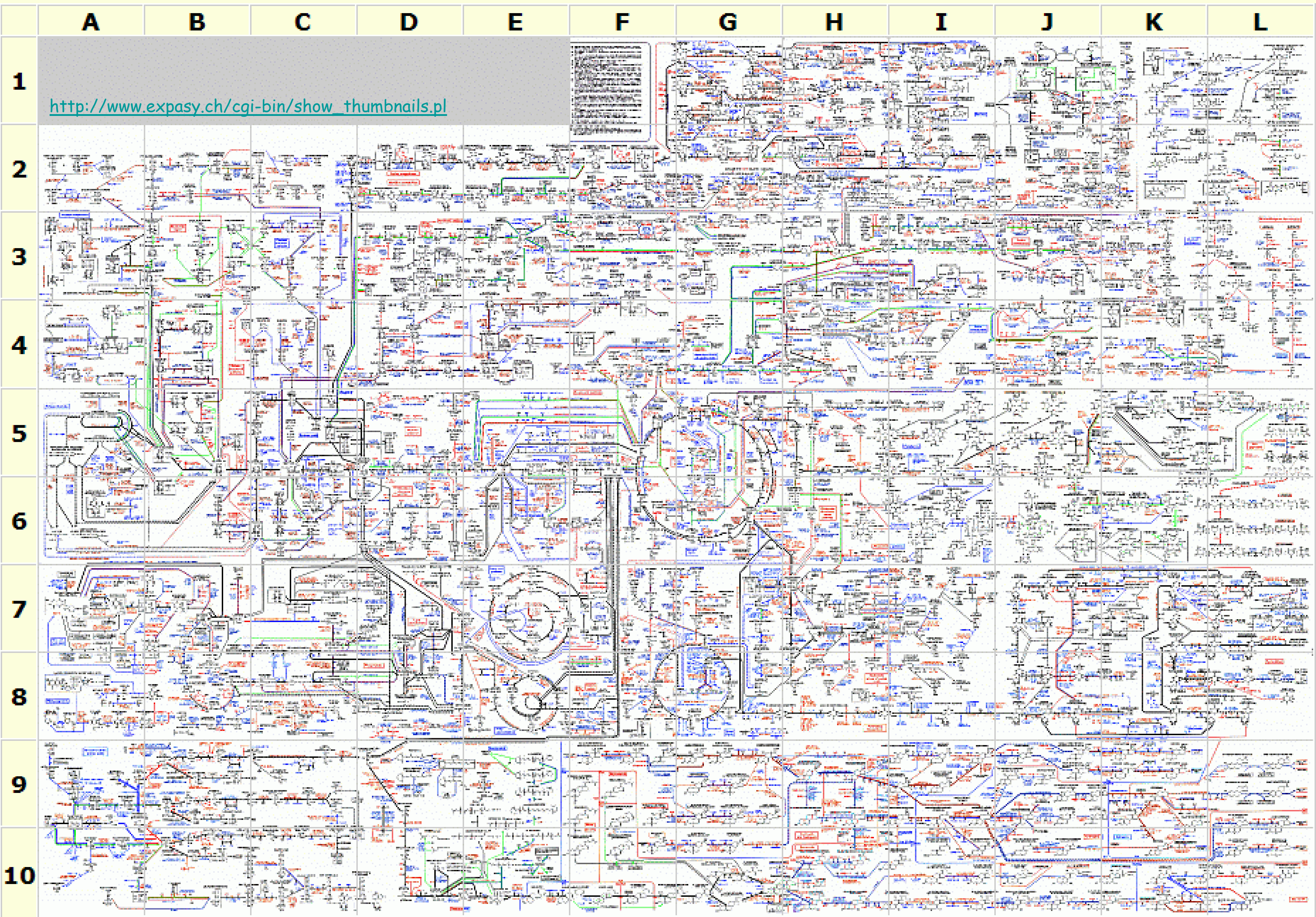
- "Stochastic":
  - Interactions have *rates*
    - Not quite discrete (hundreds or thousands of components)
    - Not quite continuous (non-trivial stochastic effects)
    - Not quite hybrid (no "switching" between regimes)

- 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].

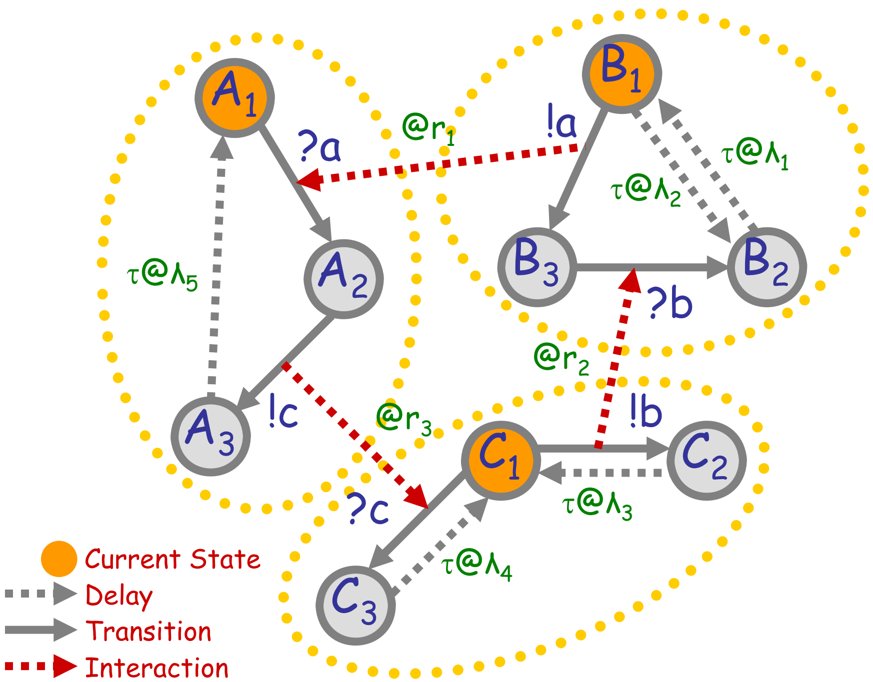
State Transitions



Compositionality (NOT!)



Interacting Automata



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

Communication channels

A_1 = ?a; A_2
A_2 = !c; A_3
A_3 = τ@λ_5; A_1

B_1 = τ@λ_2; B_2 + !a; B_3
B_2 = τ@λ_1; B_1
B_3 = ?b; B_2

C_1 = !b; C_2 + ?c; C_3
C_2 = τ@λ_3; C_1
C_3 = τ@λ_4; C_2

Automata

A_1 | B_1 | C_1

The system and initial state

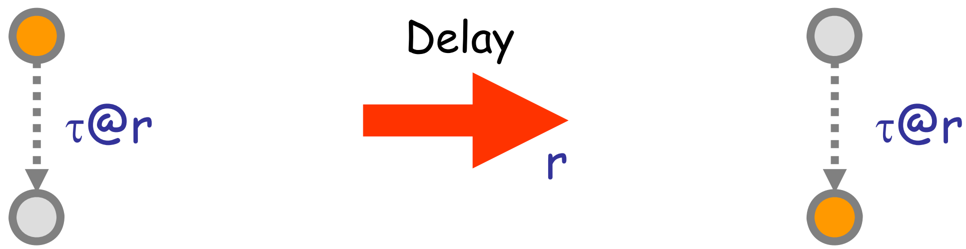
Communicating automata: a graphical FSA-like notation for "finite state restriction-free π-calculus processes". *Interacting* automata do not even exchange values on communication.

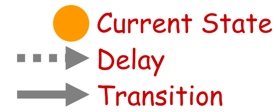
The stochastic version has *rates* on communications, and delays.

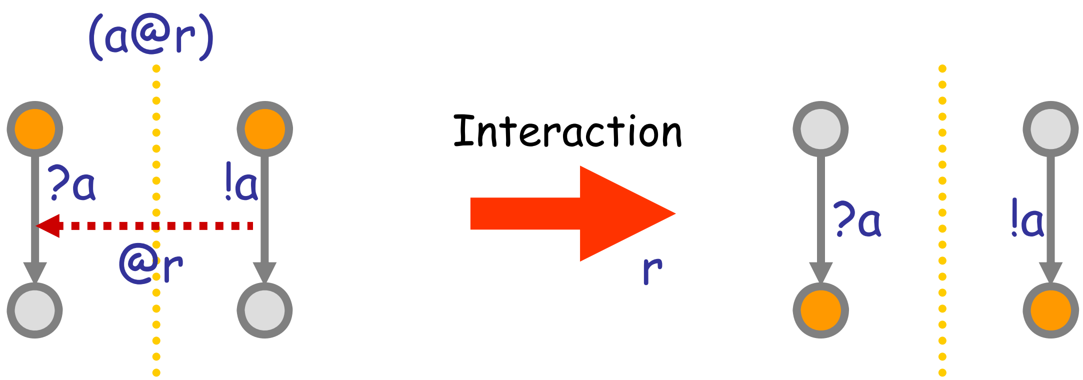
"Finite state" means: no composition or restriction inside recursion.

Analyzable by standard Markovian techniques, by first computing the "product automaton" to obtain the underlying finite Markov transition system. [Buchholz]

Interacting Automata Transition Rules







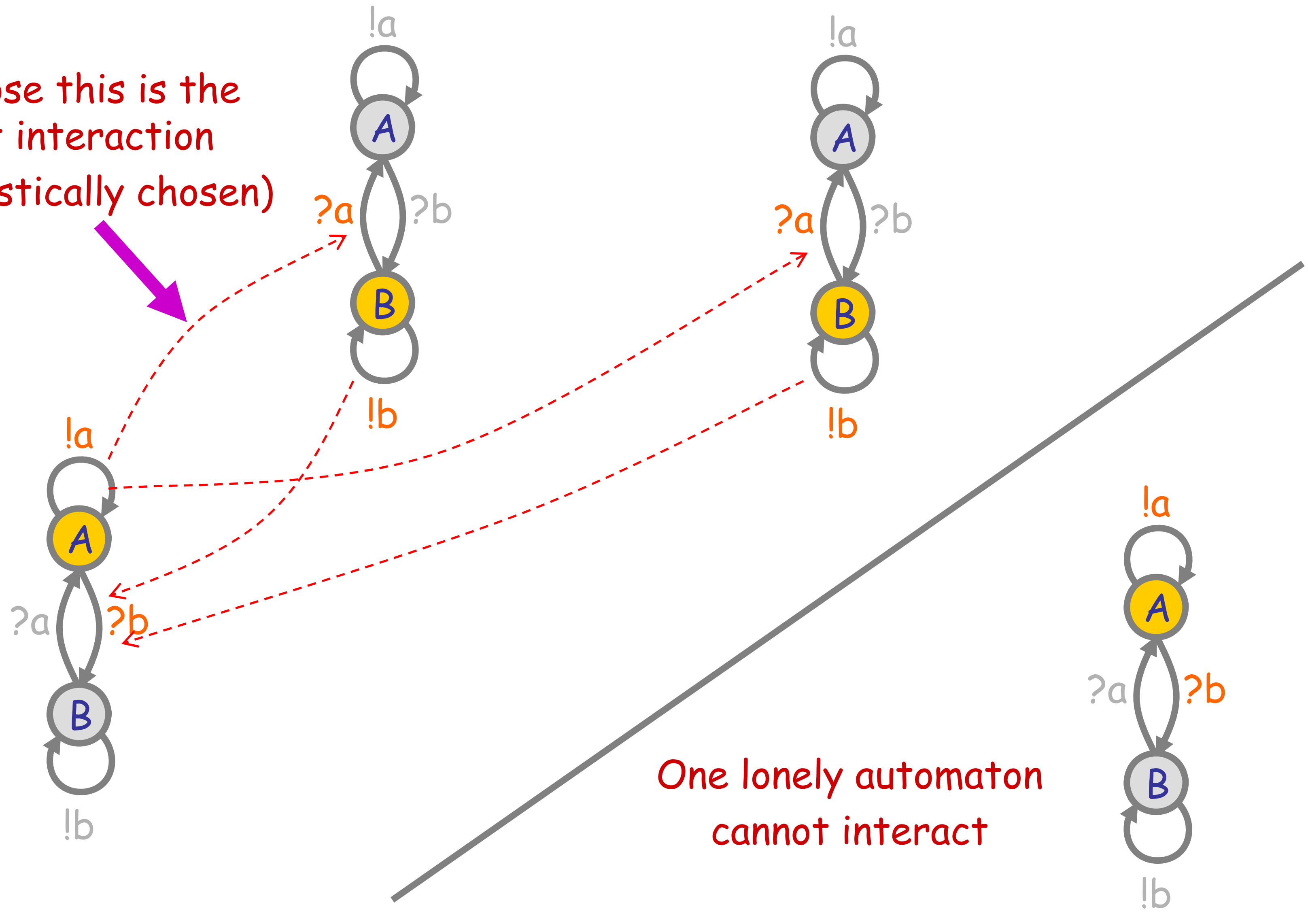
Interactions have rates. Actions DO NOT have rates.

Q: What kind of mass behavior can this produce?

(We need to understand that if want to understand biochemical systems.)

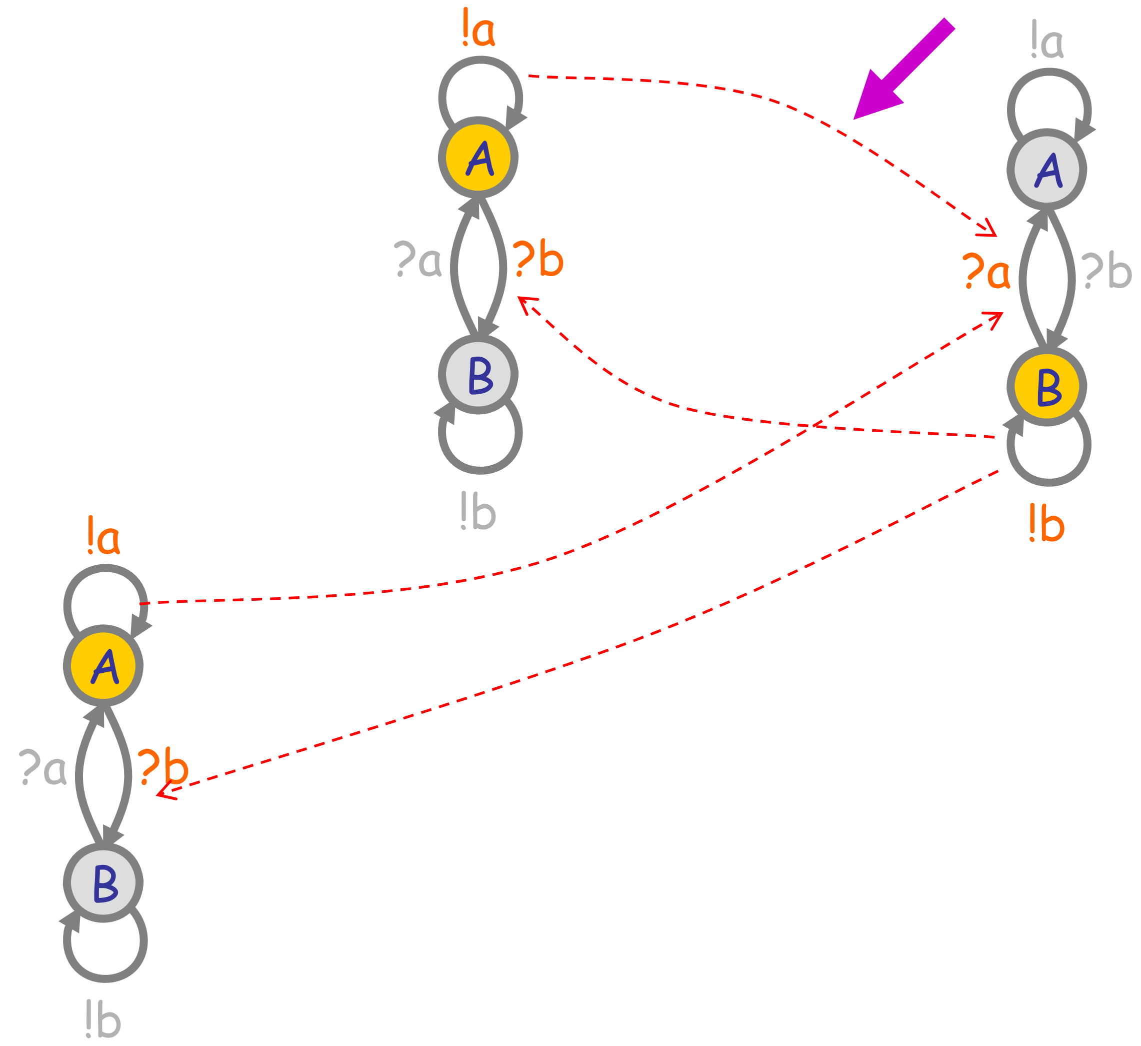
Interactions in a Population

Suppose this is the next interaction (stochastically chosen)

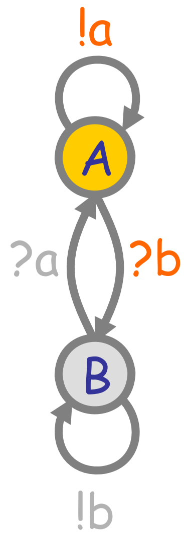


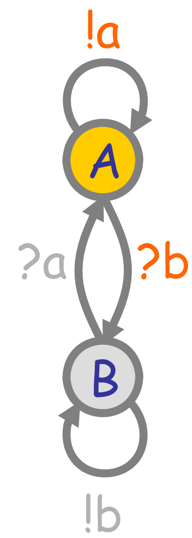
One lonely automaton cannot interact

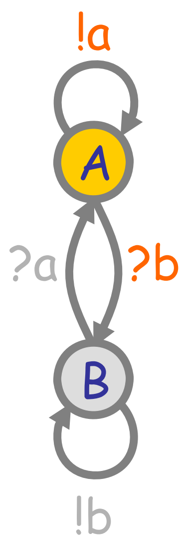
Interactions in a Population



Interactions in a Population

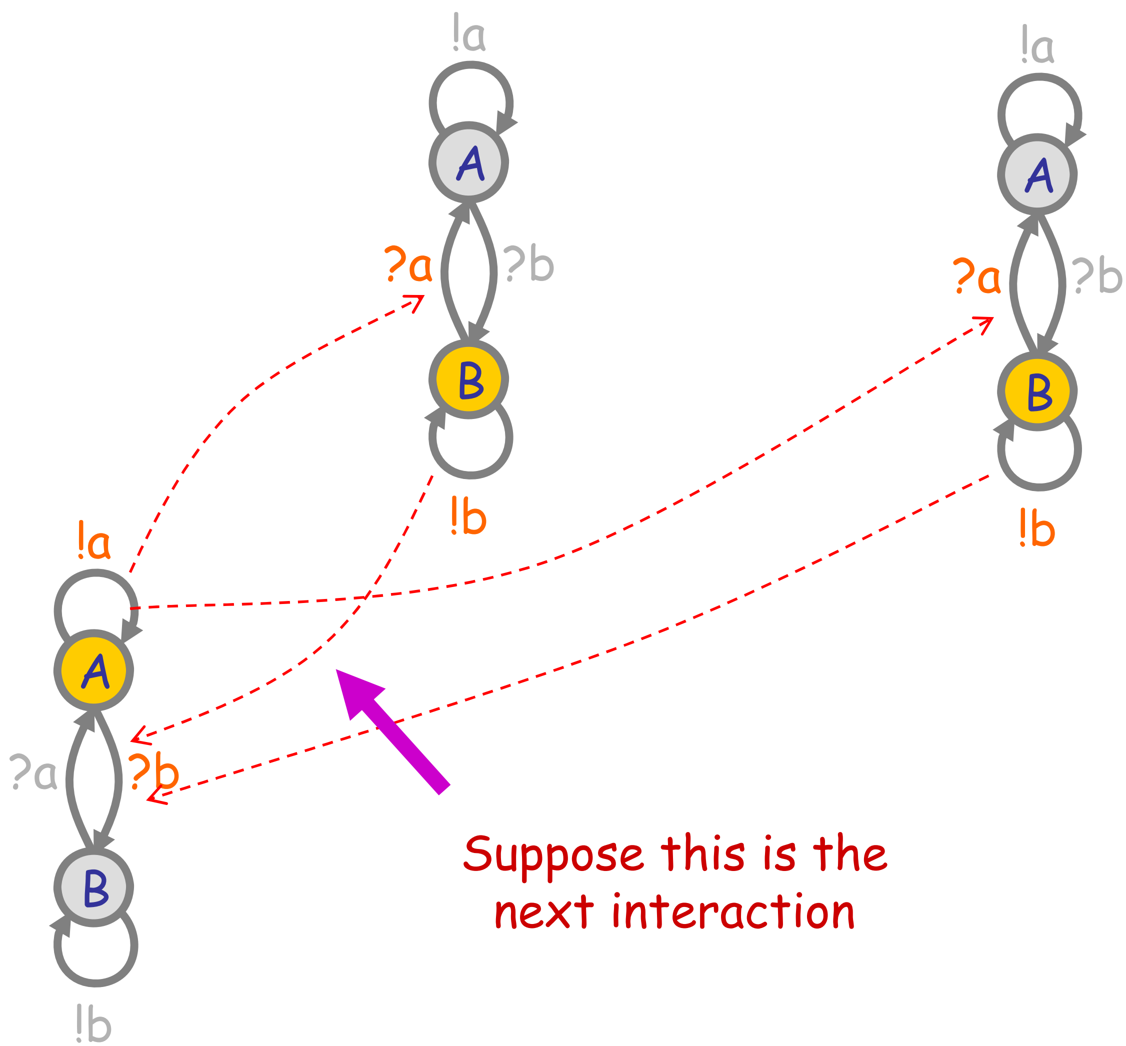




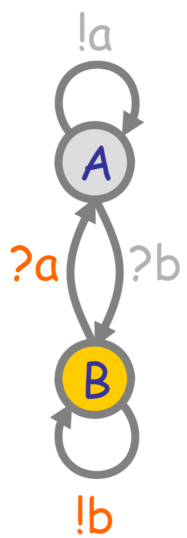


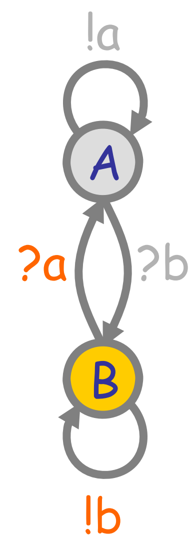
All-A stable population

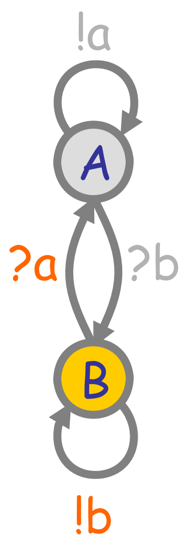
Interactions in a Population (2)



Interactions in a Population (2)



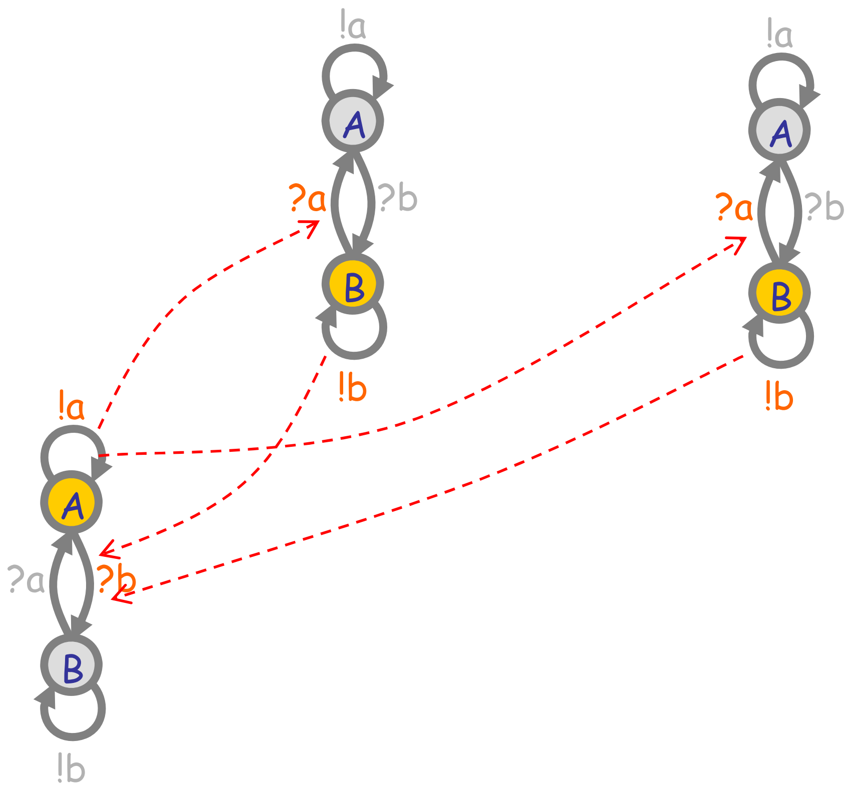


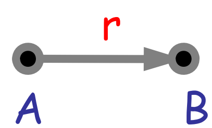


All-B stable population

Nondeterministic population behavior ("multistability")

CTMC Semantics



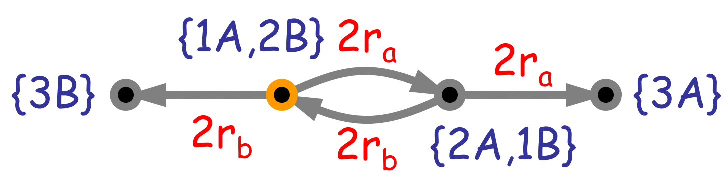


CTMC
(homogeneous) Continuous Time Markov Chain
- directed graph with no self loops
- nodes are system states
- arcs have transition rates

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



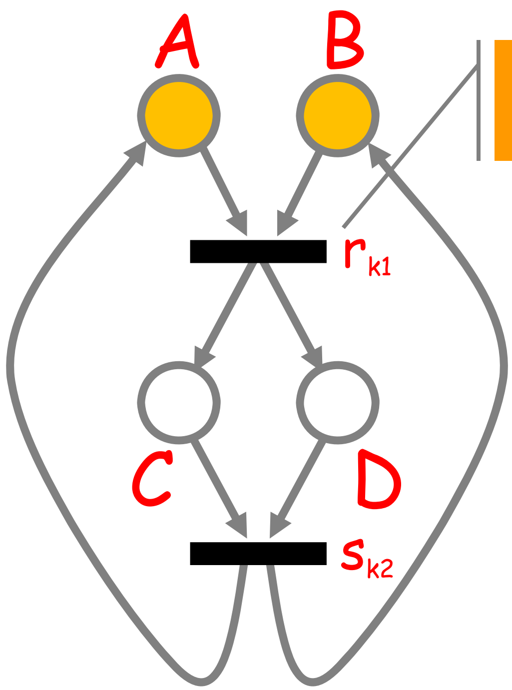
CTMC

Chemistry vs. Automata

A process calculus (chemistry)

$$r:\ A + B \to_{k1} C + D$$
$$s:\ C + D \to_{k2} A + B$$

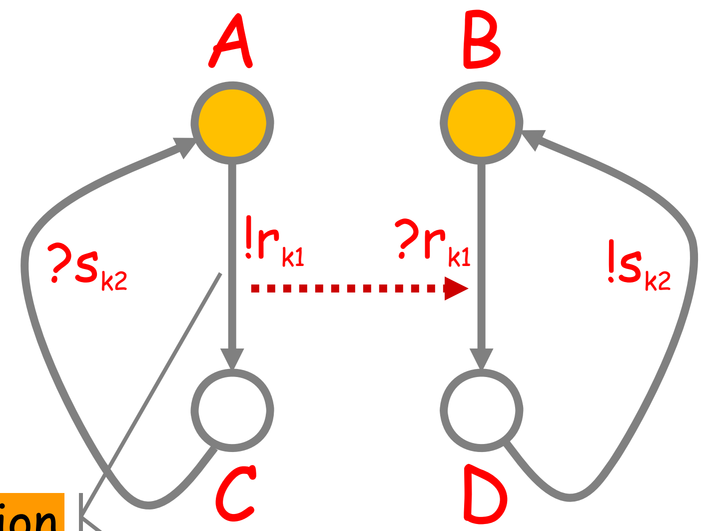
Does A become C or D?



Reaction oriented

1 line per reaction

A different process calculus (automata)



Interaction oriented

1 line per component

$$A = !r_{k1};\ C$$
$$C = ?s_{k2};\ A$$
$$B = ?r_{k1};\ D$$
$$D = !s_{k2};\ B$$

A becomes C not D!

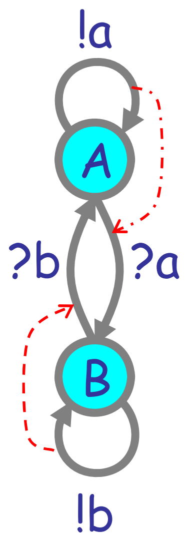
The same "model"

Maps to a CTMC	Maps to a CTMC

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.

Groupies and Celebrities



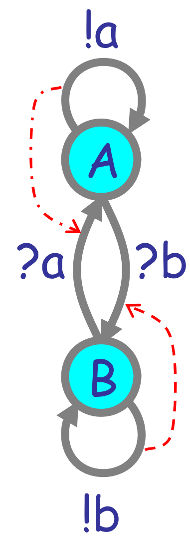
Celebrity

(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())
```

a@1.0

b@1.0



Groupie

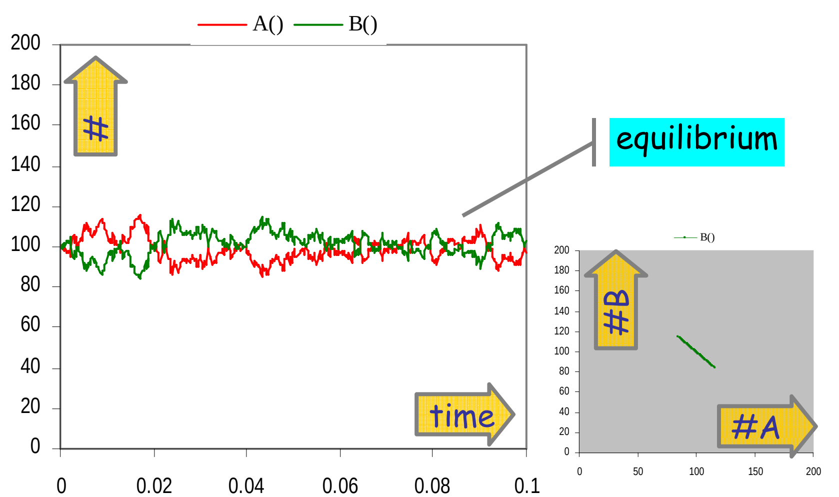
(wants to be like somebody different)

```
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 ?b; B()
and B() = do !b; B() or ?a; A()

run 100 of (A() | B())
```

a@1.0

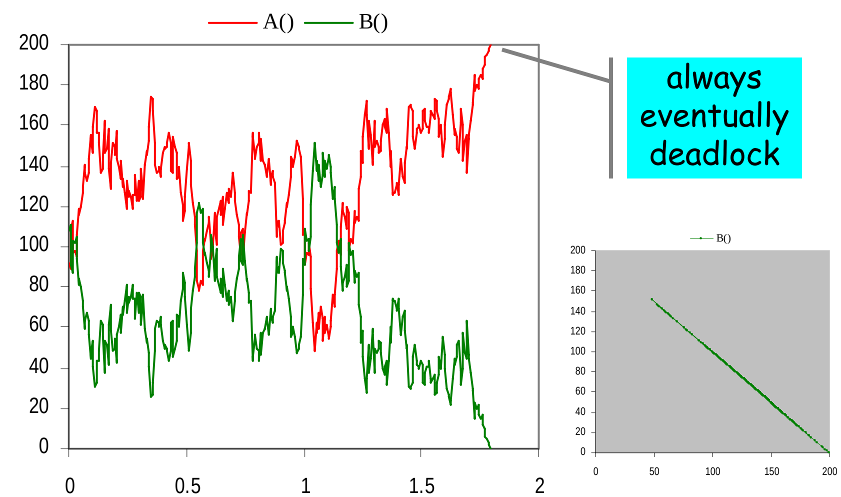
b@1.0

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.

A stochastic collective of groupies:

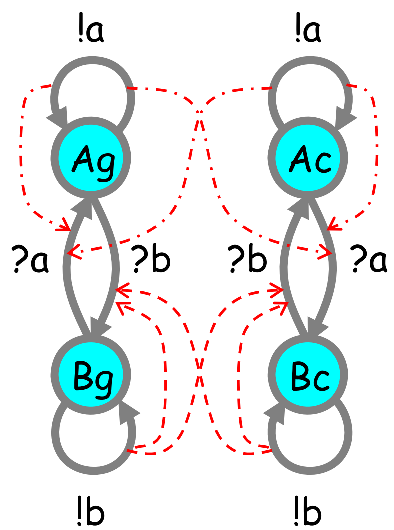


Unstable because within an A majority, an A has difficulty finding a B to emulate, but the few B's have plenty of A's to emulate, so the majority may switch to B. Leads to deadlock when everybody is in the same state and there is nobody different to emulate.

Both Together

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

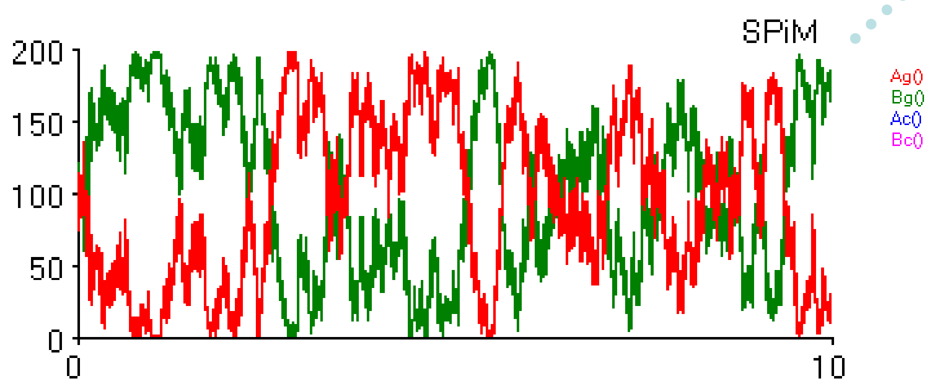
Many Groupies



A few Celebrities

```
directive sample 10.0
directive plot Ag(); Bg(); Ac(); Bc()

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

let Ac() = do !a; Ac() or ?a; Bc()
and Bc() = do !b; Bc() or ?b; Ac()

let Ag() = do !a; Ag() or ?b; Bg()
and Bg() = do !b; Bg() or ?a; Ag()

run 1 of Ac()
run 100 of (Ag() | Bg())
```

never deadlock

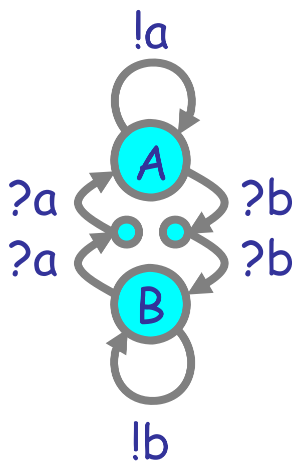


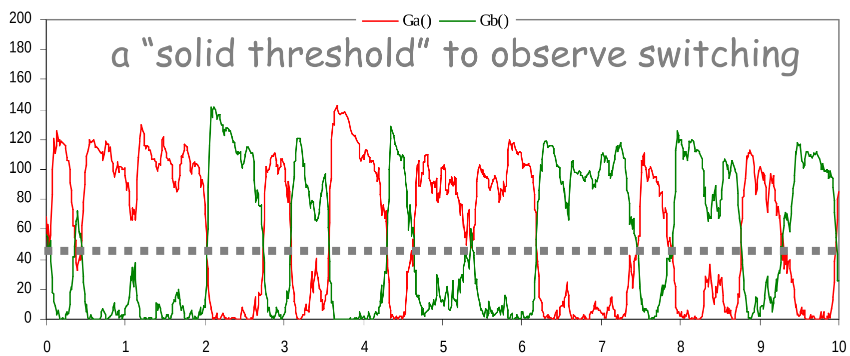
A tiny bit of "noise" can make a huge difference

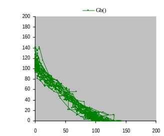
Regularity can arise not far from chaos

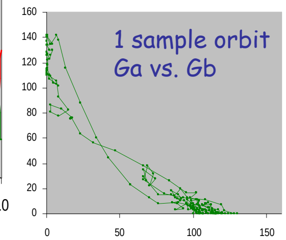
Hysteric Groupies

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

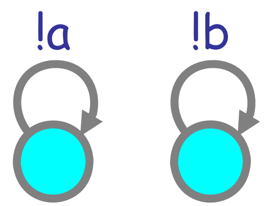








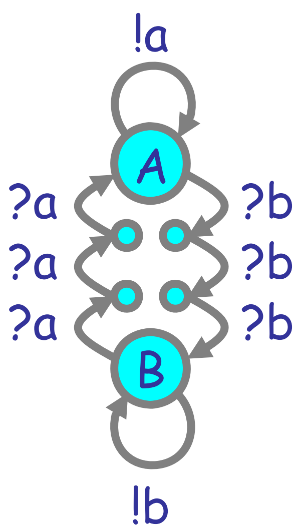

```
directive sample 10.0 1000
directive plot Ga(); Gb()

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

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

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

run 100 of (Ga() | Gb())
run   1 of (Da() | Db())
```

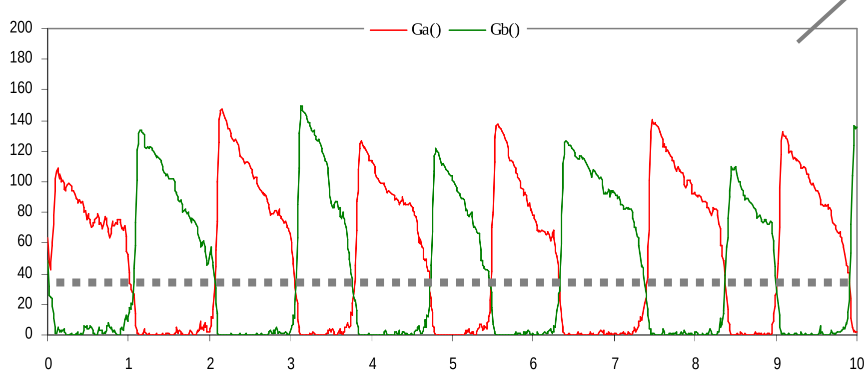


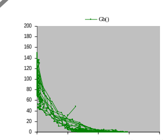
(With doping to break deadlocks)

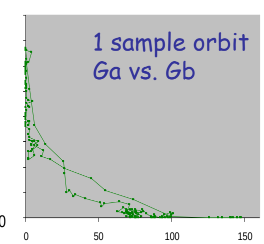
N.B.: It will not oscillate without doping (noise)

"regular" oscillation





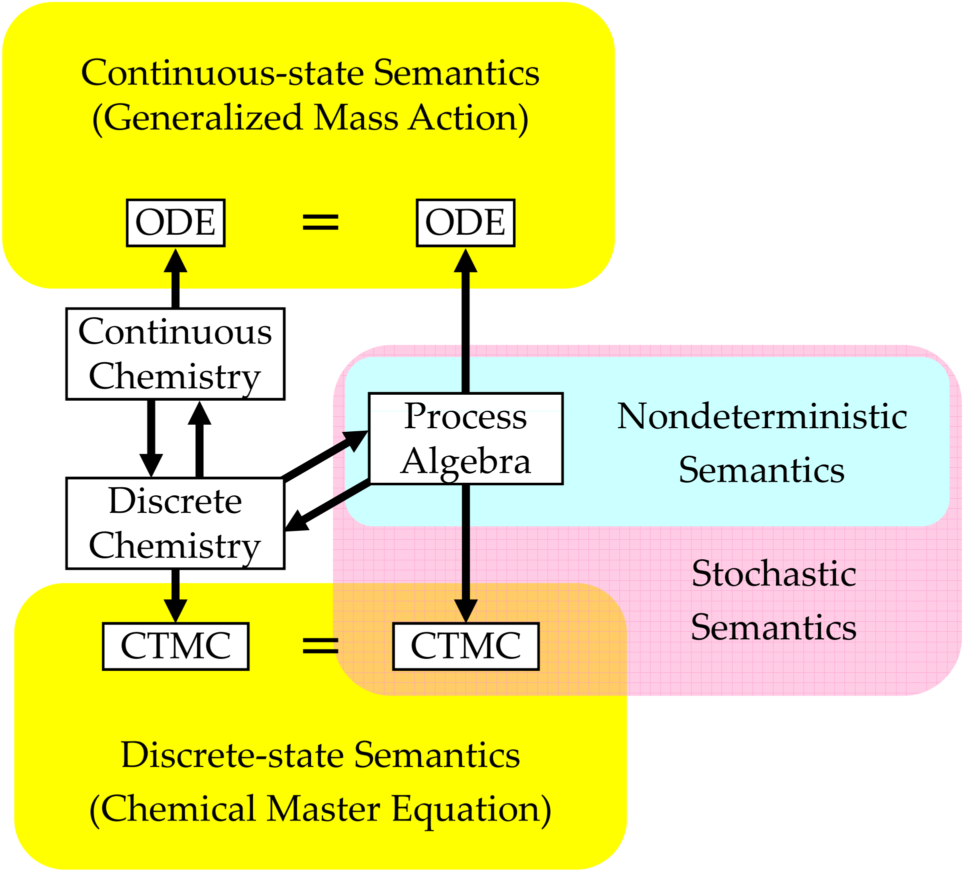





```
directive sample 10.0 1000
directive plot Ga(); Gb()

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

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

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

run 100 of (Ga() | Gb())
run   1 of (Da() | Db())
```

Semantics of Collective Behavior

The Two Semantic Faces of Chemistry



These diagrams commute via appropriate maps.

L. Cardelli: "On Process Rate Semantics"

From Processes to Chemistry

Chemical Ground Form (CGF)

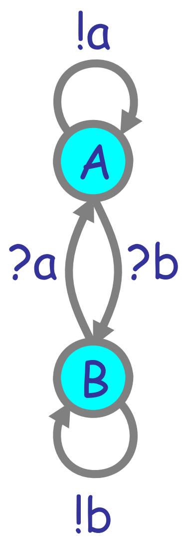
$E ::= 0 \;\vdots\; X{=}M, E$	Reagents
$M ::= 0 \;\vdots\; \pi;P \oplus M$	Molecules
$P ::= 0 \;\vdots\; X \mid P$	Solutions
$\pi ::= \tau_{(r)} \;\vdots\; ?a_{(r)} \;\vdots\; !a_{(r)}$	Interactions (delay, input, output)
$CGF ::= E,P$	Reagents plus Initial Conditions

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 $\rightarrow$, that is we may need "|" after π.)

$\oplus$ is stochastic choice (vs. + for chemical reactions)
0 is the null solution ($P|0 = 0|P = P$)
  and null molecule ($M\oplus0 = 0\oplus M = M$)
Each X in E is a distinct *species*
Each name a is assigned a fixed rate r: $a_{(r)}$



Ex: Interacting Automata
(= finite-control CGFs: they use "|" only in initial conditions):

$A = !a;A \oplus ?b;B$ — Automaton in state A

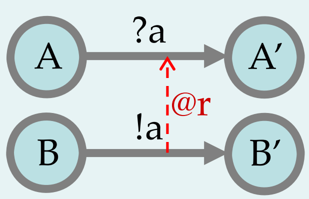
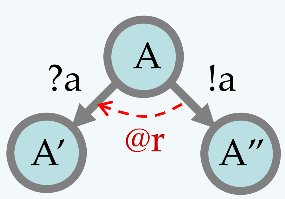
$B = !b;B \oplus ?a;A$ — Automaton in state B

$A|A|B|B$ — Initial conditions: 2A and 2B

Processes to Chemistry

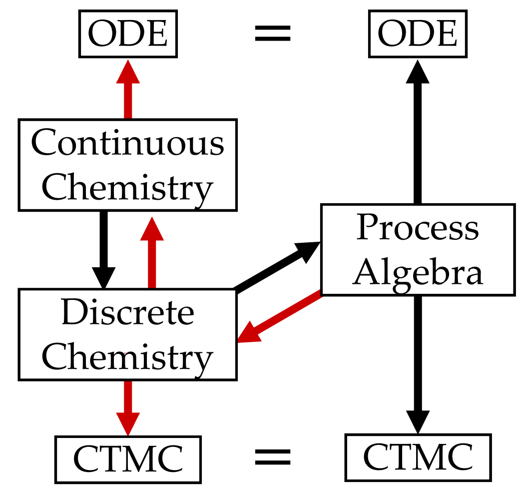
V = interaction volume

N_A = Avogadro's number

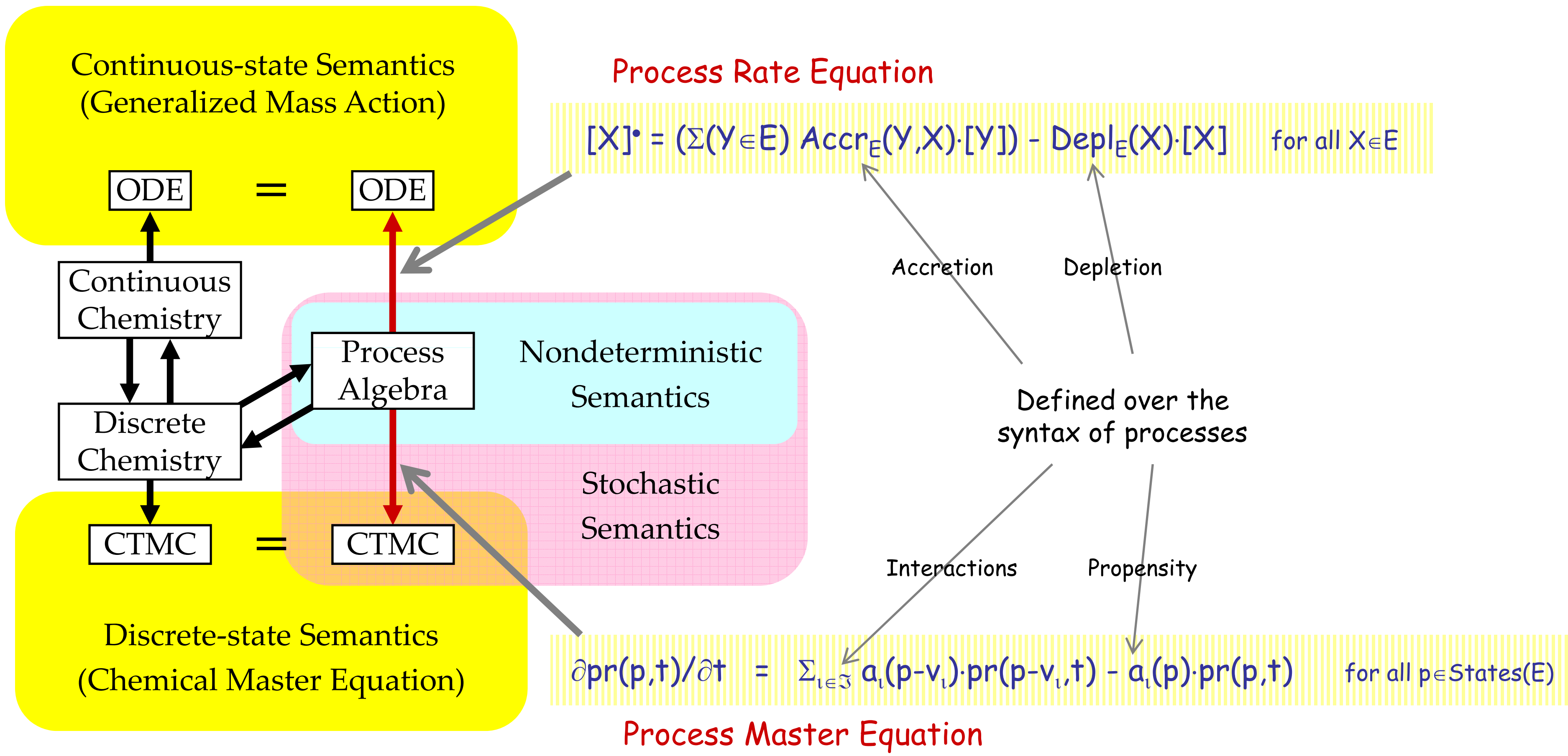
Automata →	Discrete Chemistry →	Continuous Chemistry $\gamma = N_A V$
initial states $A \mid A \mid \ldots \mid A$	initial quantities $\#A_0$	initial concentrations $[A]_0$ with $[A]_0 = \#A_0/\gamma$
	$A \xrightarrow{r} A'$	$A \rightarrow^{k} A'$ with $k = r$
	$A+B \xrightarrow{r} A'+B'$	$A+B \rightarrow^{k} A'+B'$ with $k = r\gamma$
	$A+A \xrightarrow{2r} A'+A''$	$A+A \rightarrow^{2k} A'+A''$ with $k = r\gamma/2$
	↓ CTMC	↓ ODE

Think $\gamma = 1$

i.e. $V = 1/N_A$



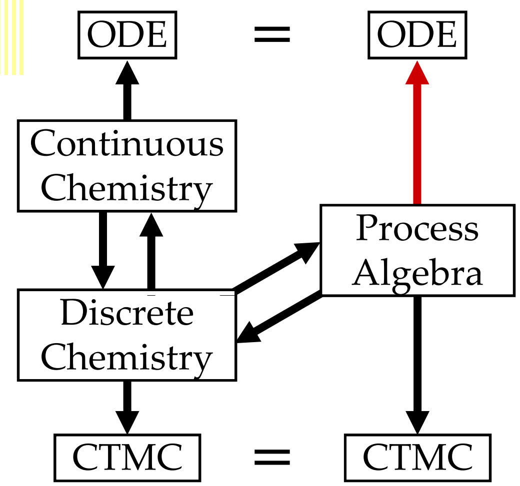
Quantitative Process Semantics



Processes to GMA Directly

Process Rate Equation for Reagents E

$$[X]^{\bullet} = (\Sigma(Y{\in}E)\ Accr_E(Y,X)\cdot[Y]) - Depl_E(X)\cdot[X] \qquad \text{for all } X{\in}E$$



$Depl_E(X) =$
- $\Sigma(i{:}\ E.X.i{=}\tau_{(r)};P)\ r\ +$
- $\Sigma(i{:}\ E.X.i{=}?a_{(r)};P)\ r\gamma\cdot OutsOn_E(a)\ +$
- $\Sigma(i{:}\ E.X.i{=}!a_{(r)};P)\ r\gamma\cdot InsOn_E(a)$

$Accr_E(Y, X) =$
- $\Sigma(i{:}\ E.Y.i{=}\tau_{(r)};P)\ \#X(P)\cdot r\ +$
- $\Sigma(i{:}\ E.Y.i{=}?a_{(r)};P)\ \#X(P)\cdot r\gamma\cdot OutsOn_E(a)\ +$
- $\Sigma(i{:}\ E.Y.i{=}!a_{(r)};P)\ \#X(P)\cdot r\gamma\cdot InsOn_E(a)$

$InsOn_E(a) = \Sigma(Y{\in}E)\ \#\{Y.i \mid E.Y.i{=}?a_{(r)};P\}\cdot[Y]$

$OutsOn_E(a) = \Sigma(Y{\in}E)\ \#\{Y.i \mid E.Y.i{=}!a_{(r)};P\}\cdot[Y]$

$X = \tau_{(r)};0 \quad \rightarrow \quad [X]^{\bullet} = -r[X]$

$X = ?a_{(r)};0,\ Y = !a_{(r)};0 \quad \rightarrow \quad [X]^{\bullet} = -r\gamma[X][Y],\ [Y]^{\bullet} = -r\gamma[X][Y]$

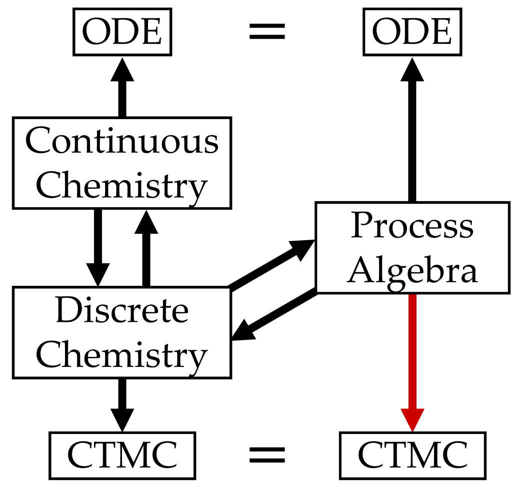
$X = ?a_{(r)};0 \oplus !a_{(r)};0 \quad \rightarrow \quad [X]^{\bullet} = -2r\gamma[X]^2$

Processes to CME Directly

Process Master Equation for Reagents E

$$\partial pr(p,t)/\partial t \;=\; \Sigma_{\iota\in\mathfrak{I}}\, a_\iota(p-v_\iota)\cdot pr(p-v_\iota,t) - a_\iota(p)\cdot pr(p,t) \qquad \text{for all } p\in States(E)$$

$pr(p,t) = Pr\{\mathbf{S}(t)=p \mid \mathbf{S}(0)=p_0\}$ is the conditional probability of the system being in state p (a multiset of molecules) at time t given that it was in state p_0 at time 0.



$\mathfrak{I} = \{\{X.i\}$ *s.t.* $E.X.i = \tau_{(r)};Q\} \cup$
$\{\{X.i, Y.j\}$ *s.t.* $E.X.i = ?n_{(r)};Q$ and $E.Y.j = !n_{(r)};R\}$
is the set of possible interactions in E

v_ι is the *state change* caused by an interaction $\iota\in\mathfrak{I}$.

$v_\iota = -X+Q$ if $\iota = \{X.i\}$ *s.t.* $E.X.i = \tau_{(r)};Q$
$v_\iota = -X-Y+Q\_R$ if $\iota = \{X.i, Y.j\}$ *s.t.* $E.X.i = ?n_{(r)};Q$ and $E.Y.j = !n_{(r)};R$

a_ι is the *propensity* of interaction ι in state p. Here $p^{\#X}$ is the number of X in p.

$a_\iota(p) = r\cdot p^{\#X}$ if $\iota = \{X.i\}$ *s.t.* $E.X.i = \tau_{(r)};Q$
$a_\iota(p) = r\cdot p^{\#X}\cdot p^{\#Y}$ if $\iota = \{X.i, Y.j\}$ *s.t.* $X\neq Y$ and $E.X.i = ?a_{(r)};Q$ and $E.Y.j = !a_{(r)};R$
$a_\iota(p) = r\cdot p^{\#X}\cdot(p^{\#X}-1)$ if $\iota = \{X.i, X.j\}$ *s.t.* $E.X.i = ?a_{(r)};Q$ and $E.X.j = !a_{(r)};R$

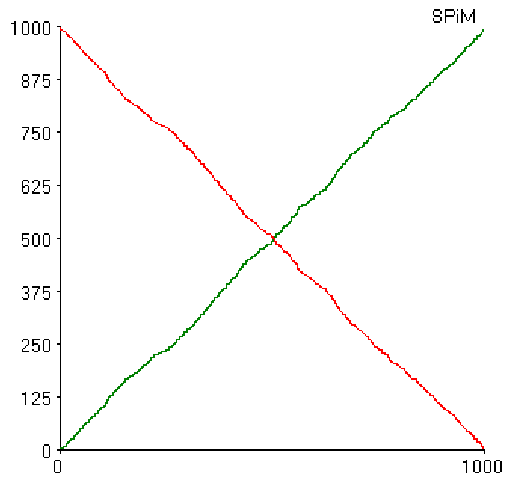
Examples of stochastic collectives where:

(1) Simulation is puzzling and ODE analysis is more useful.

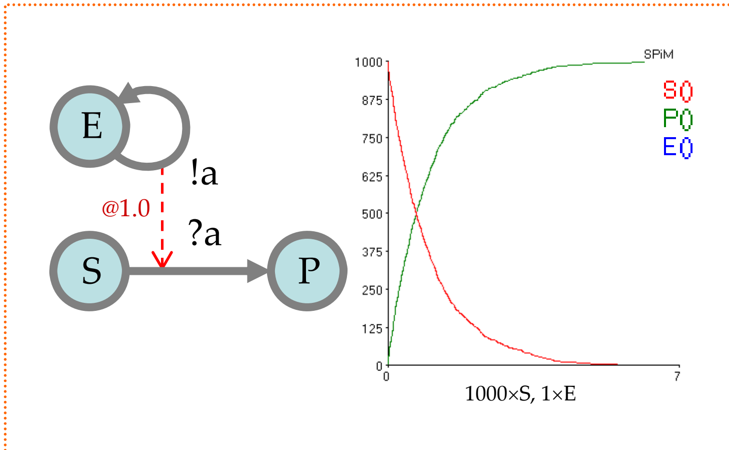
(2) ODE analysis is puzzling and simulation is more useful.

Zero-Order Regime

Or: build me a process like this:



Second-order and Zero-order Regime

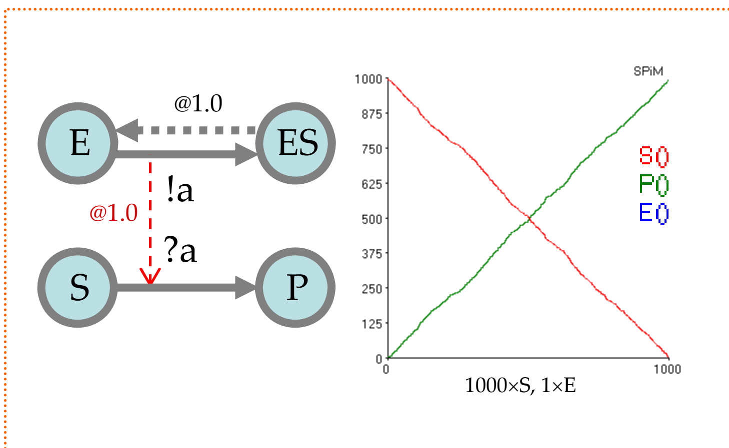


$$E+S \to^{r} E+P$$

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

new a@1.0:chan()

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

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

Second-Order Regime

$[S]^\bullet = -r[E][S]$

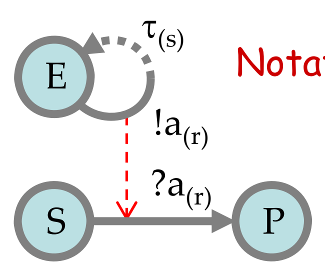


$$E+S \to^{r} ES+P$$
$$ES \to^{s} E$$

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

new a@1.0:chan()

let E() = !a; delay@1.0; E()
and S() = ?a; P()
and P() = ()

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

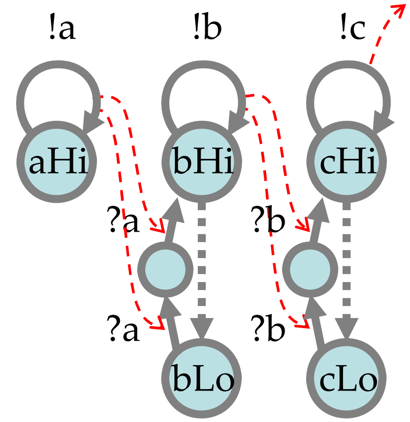
Zero-Order Regime

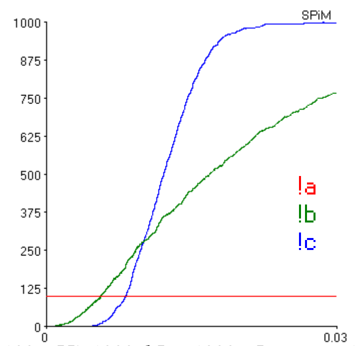
$[S]^\bullet \cong -1$ (by assuming $[ES]^\bullet = 0$)



Notation

Cascades



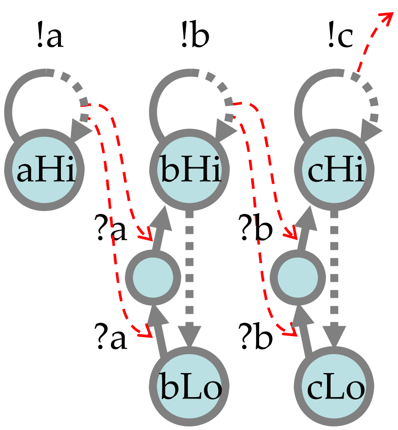


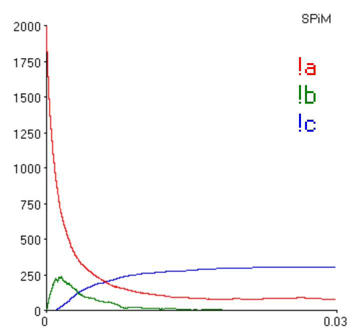
100×aHi, 1000×bLo, 1000×cLo, rates=1.0

Second-Oder Regime cascade:
a signal amplifier (MAPK)

aHi > 0 ⇒ cHi = max

```
directive sample 0.03
directive plot !a; !b; !c

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

let Amp_hi(a:chan, b:chan) =
  do !b; Amp_hi(a,b) or delay@1.0; Amp_lo(a,b)
and Amp_lo(a:chan, b:chan) =
  ?a; ?a; Amp_hi(a,b)

run 1000 of (Amp_lo(a,b) | Amp_lo(b,c))

let A() = !a; A()
run 100 of A()
```





2000×aHi, 1000×bLo, 1000×cLo, rates=1.0

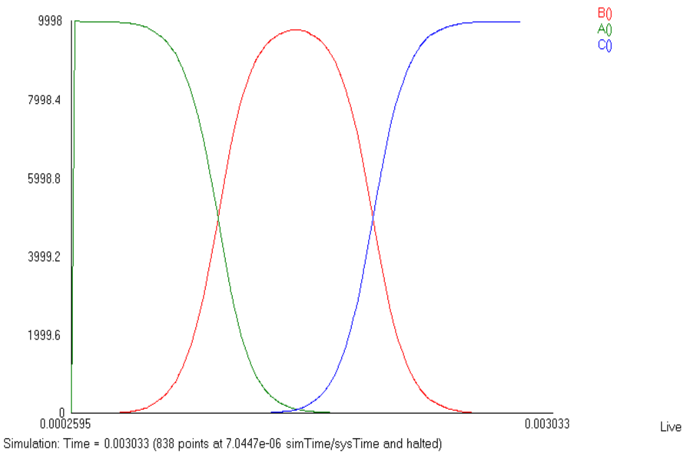
Zero-Oder Regime cascade:
a signal *divider!*

aHi = max ⇒ cHi = 1/3 max

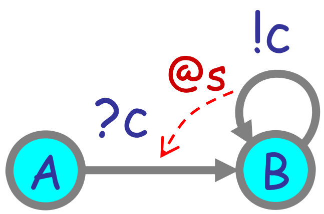
```
directive sample 0.03
directive plot !a; !b; !c

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

let Amp_hi(a:chan, b:chan) =
  do !b; delay@1.0; Amp_hi(a,b) or delay@1.0; Amp_lo(a,b)
and Amp_lo(a:chan, b:chan) =
  ?a; ?a; Amp_hi(a,b)

run 1000 of (Amp_lo(a,b) | Amp_lo(b,c))

let A() = !a; delay@1.0; A()
run 2000 of A()
```

Waves

Or: build me a process like this:



Nonlinear Transition (NLT)



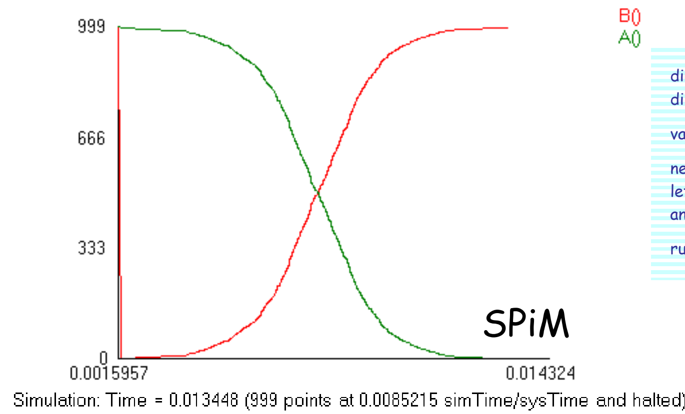
$A = ?c_{(s)};B$

$B = !c_{(s)};B$

$A{+}B \rightarrow^{s} B{+}B$

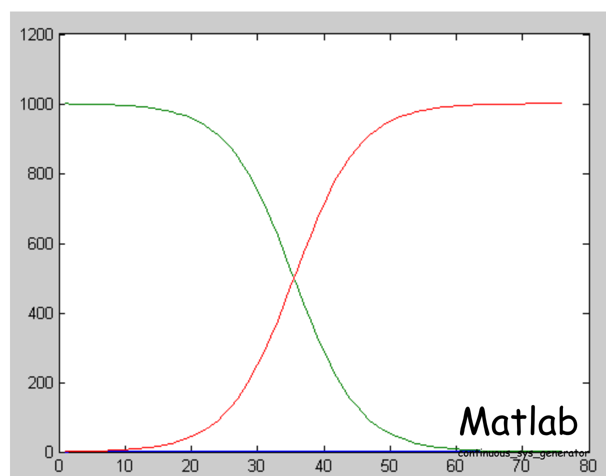
$[A]^{\bullet} = -s[A][B]$

$[B]^{\bullet} = s[A][B]$



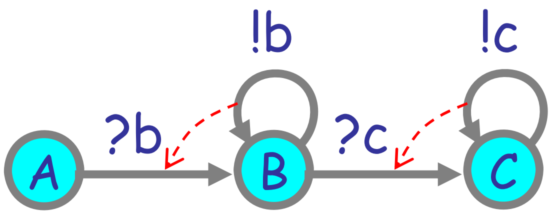

```
directive sample 0.02 1000
directive plot B(); A()

val s=1.0

new c@s:chan
let A() = ?c; B()
and B() = !c;B()

run (1000 of A() | 1 of B())
```

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




```
interval/step [0:0.001:0.0]
(A)    dx1/dt = - x1*x2      1000.0
(B)    dx2/dt = x1*x2        1.0
```

Two NLTs: Bell Shape



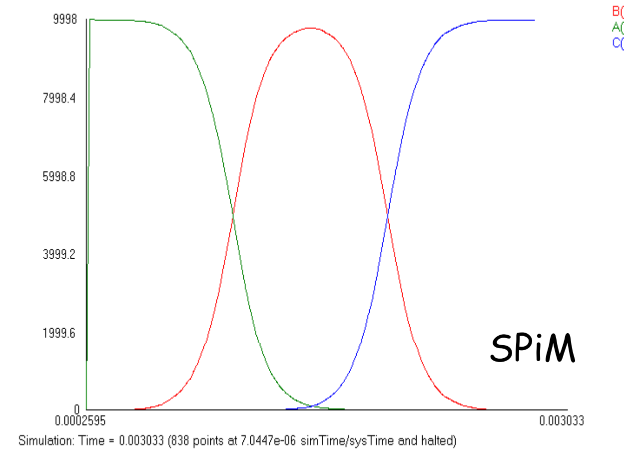
$[B]^{\bullet} = [B]([A]-[C])$

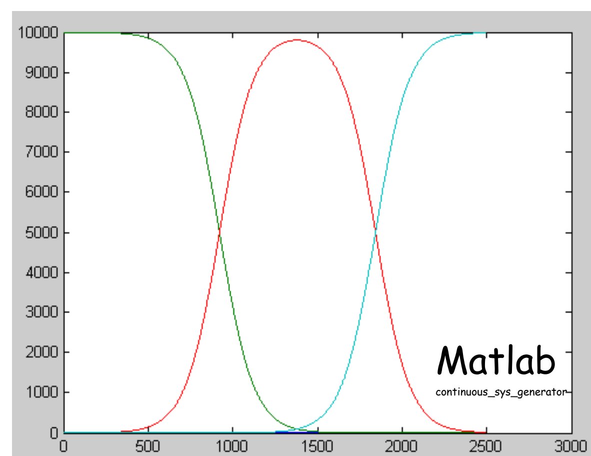
```
directive sample 0.0025 1000
directive plot B(); A(); C()

new b@1.0:chan new c@1.0:chan

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

run ((10000 of A()) | B() | C())
```

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

$$A+B \rightarrow^{1} B+B$$
$$B+C \rightarrow^{1} C+C$$

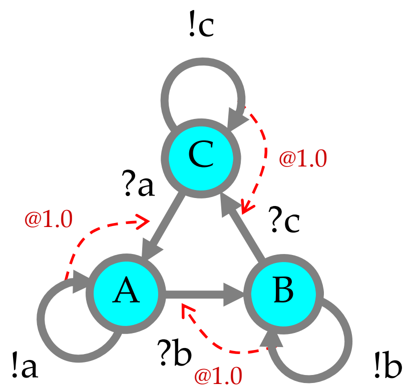
$$[A]^{\bullet} = -[A][B]$$
$$[B]^{\bullet} = [A][B]-[B][C]$$
$$[C]^{\bullet} = [B][C]$$

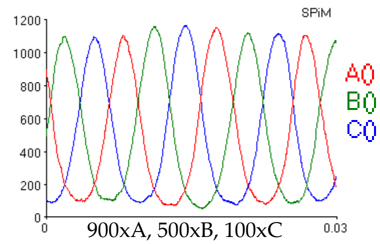





```
interval/step [0:0.000001:0.0025]
(A)    dx1/dt = -x1*x2             10000.0
(B)    dx2/dt = x1*x2 - x2*x3      1.0
(C)    dx3/dt = x2*x3              1.0
```

NLT in a Cycle: Oscillator






```
directive sample 0.03 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 ?b; B()
and B() = do !b;B() or ?c; C()
and C() = do !c;C() or ?a; A()

run (900 of A() | 500 of B() | 100 of C())
```

$A = !a_{(s)};A \oplus ?b_{(s)};B$

$B = !b_{(s)};B \oplus ?c_{(s)};C$

$C = !c_{(s)};C \oplus ?a_{(s)};A$

$A+B \to^s B+B$

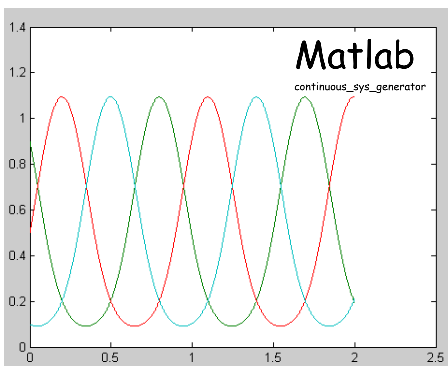
$B+C \to^s C+C$

$C+A \to^s A+A$

$[A]^\bullet = -s[A][B]+s[C][A]$

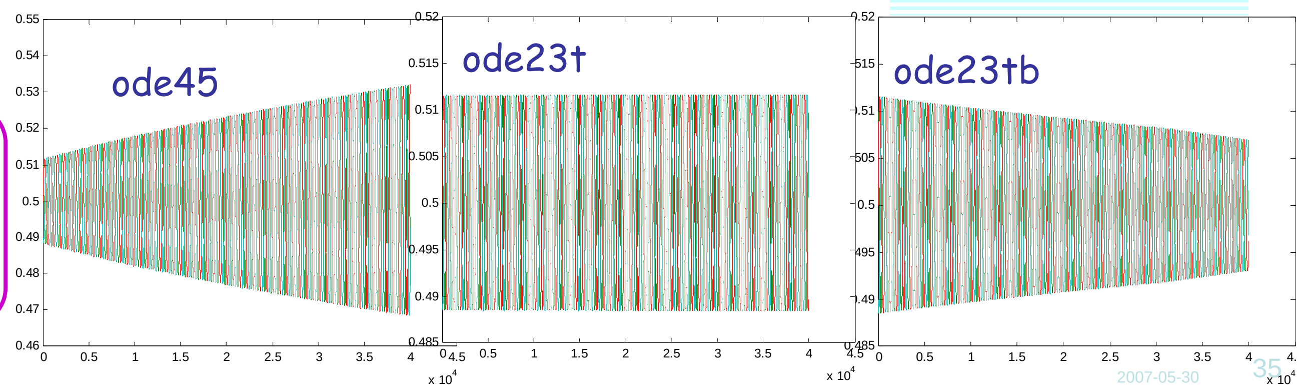
$[B]^\bullet = -s[B][C]+s[A][B]$

$[C]^\bullet = -s[C][A]+s[B][C]$




```
interval/step [0:0.001:20.0]
(A)   dx1/dt = - x1*x2 + x3*x1    0.9
(B)   dx2/dt = - x2*x3 + x1*x2    0.5
(C)   dx3/dt = - x3*x1 + x2*x3    0.1
```

```
interval/step [0:0.01:400.0]
(A)   dx1/dt = - x1*x2 + x3*x1    0.51
(B)   dx2/dt = - x2*x3 + x1*x2    0.5
(C)   dx3/dt = - x3*x1 + x2*x3    0.49
```



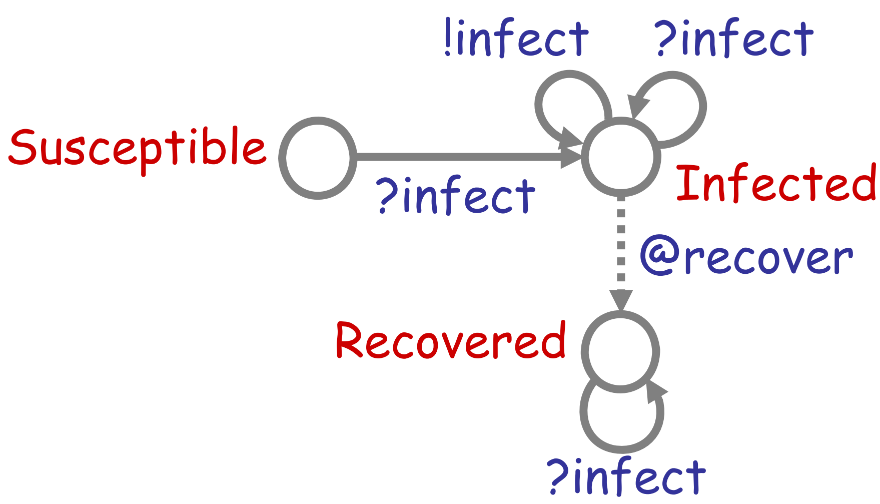
Epidemics

Beyond Chemical Interactions

Kermack, W. O. and McKendrick, A. G. "**A Contribution to the Mathematical Theory of Epidemics.**" *Proc. Roy. Soc. Lond. A* **115**, 700-721, 1927.

http://mathworld.wolfram.com/Kermack-McKendrickModel.html

Epidemics



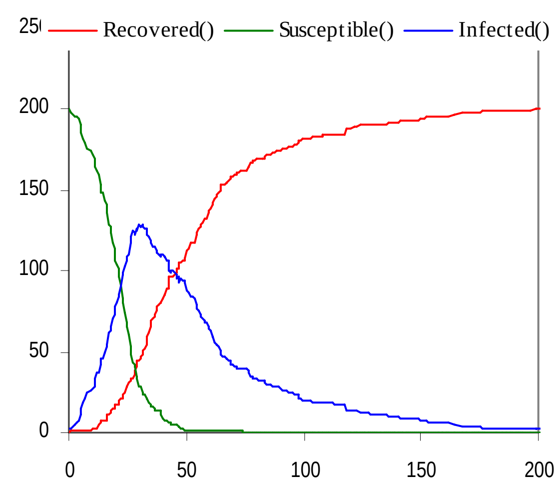

```
directive sample 500.0 1000
directive plot Recovered(); Susceptible(); Infected()

new infect @0.001:chan()
val recover = 0.03

let Recovered() =
  ?infect; Recovered()

and Susceptible() =
  ?infect; Infected()

and Infected() =
  do !infect; Infected()
  or ?infect; Infected()
  or delay@recover; Recovered()

run (200 of Susceptible() | 2 of Infected())
```

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 defined the model.



ODE

$S = ?i_{(t)};I$
$I = !i_{(t)};I \oplus ?i_{(t)};I \oplus \tau_r;R$
$R = ?i_{(t)};R$

Differentiating Processes!

$S + I \rightarrow^{t\gamma} I + I$
$I + I \rightarrow^{t\gamma} I + I$
$I \rightarrow^{r} R$
$R + I \rightarrow^{t\gamma} R + I$

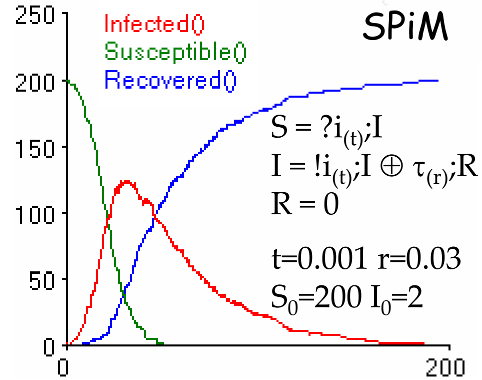
"useless" reactions

$[S]^\bullet = -t\gamma[S][I]$
$[I]^\bullet = t\gamma[S][I]-r[I]$
$[R]^\bullet = r[I]$

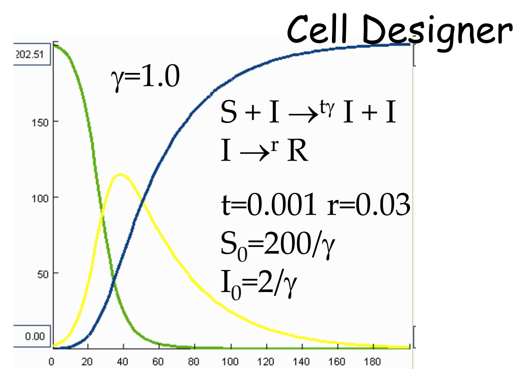
Automata produce the standard ODEs!

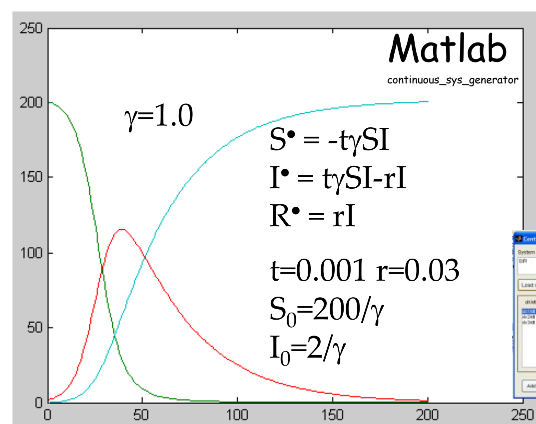
$$\frac{dS}{dt} = -aIS$$
$$\frac{dI}{dt} = aIS - bI$$
$$\frac{dR}{dt} = bI$$

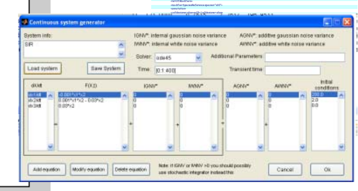
(the Kermack-McKendrick, or SIR model)



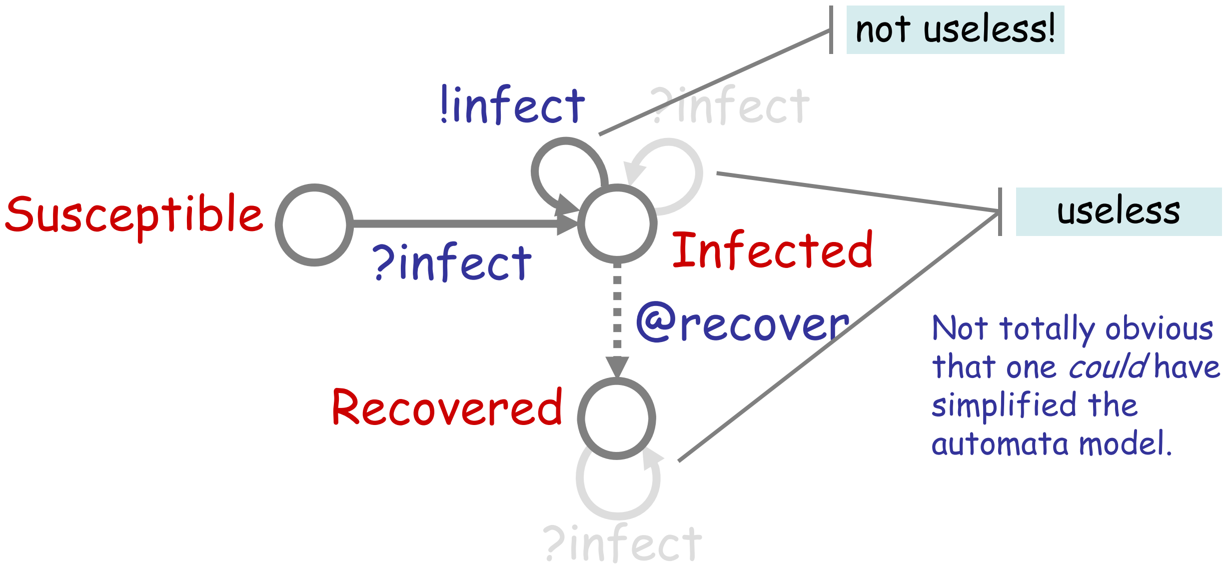

```
directive sample 500.0 1000
directive plot Recovered(); Susceptible(); Infected()

new infect @0.001:chan()
val recover = 0.03

let Recovered() =
  ?infect; Recovered()

and Susceptible() =
  ?infect; Infected()

and Infected() =
  do !infect; Infected()
  or ?infect; Infected()
  or delay@recover; Recovered()

run (200 of Susceptible() | 2 of Infected())
```







Simplified Model



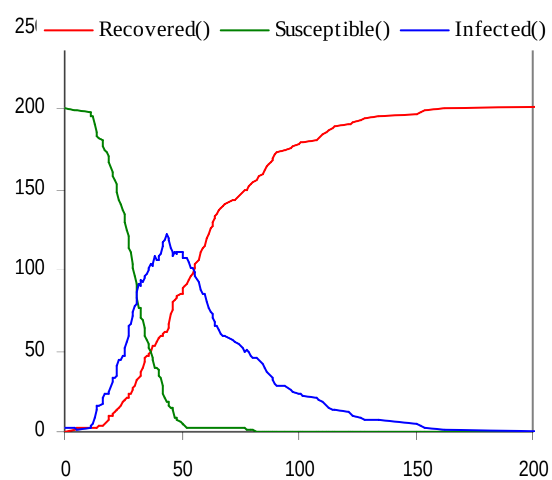
Not totally obvious that one *could* have simplified the automata model.

$$S = ?i_{(t)};I$$
$$I = !i_{(t)};I \oplus \tau_r;R$$
$$R = 0$$

$$S + I \to^{t\gamma} I + I$$
$$I \to^{r} R$$

$$[S]^{\bullet} = -t\gamma[S][I]$$
$$[I]^{\bullet} = t\gamma[S][I]-r[I]$$
$$[R]^{\bullet} = r[I]$$

```
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())
```

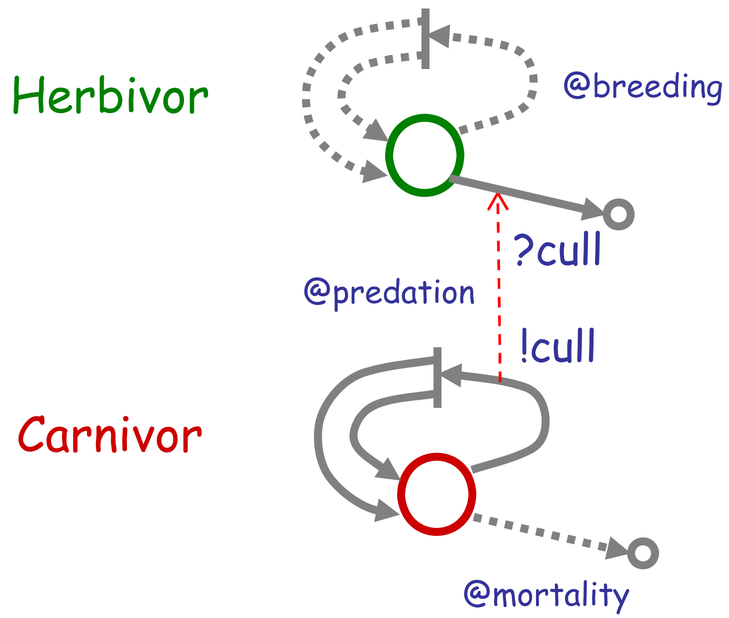


Same ODE, hence equivalent automata models.

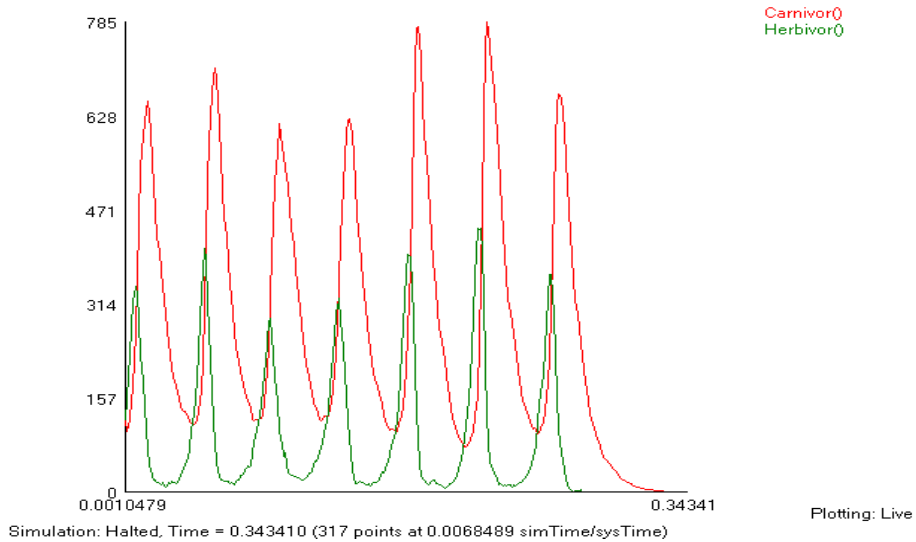
Lotka-Volterra

Beyond Automata

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!

Lotka-Volterra in Matlab

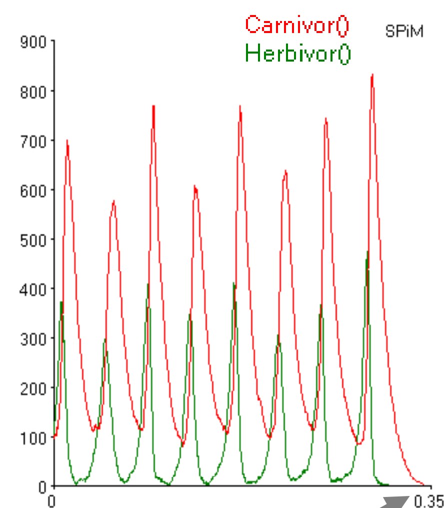
$H = \tau_b; (H|H) \oplus ?c_{(p)};0$
$C = \tau_m;0 \oplus !c_{(p)};(C|C)$
$\#H_0, \#C_0$

$H \to^{b} H + H$
$C \to^{m} 0$
$H + C \to^{p\gamma} C + C$
$[H]_0 = \#H_0/\gamma$
$[C]_0 = \#C_0/\gamma$

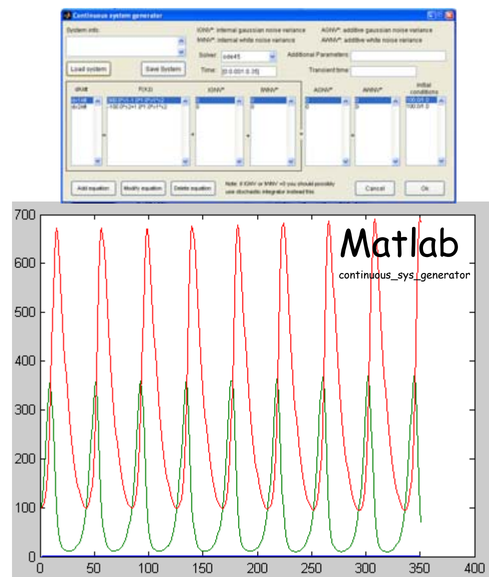
$[H]^\bullet = b[H]-p\gamma[H][C]$
$[C]^\bullet = -m[C]+p\gamma[H][C]$
$[H]_0 = \#H_0/\gamma$
$[C]_0 = \#C_0/\gamma$

$m=100.0$
$b=300.0$
$p=1.0$
$\gamma=1.0$
$\#H_0 = 100$
$\#C_0 = 100$

```
directive sample 0.35 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()
```



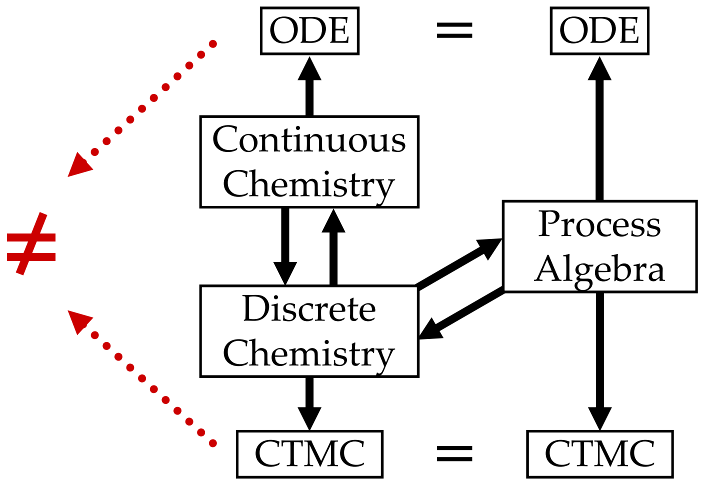
Extinction



No extinction

Which one is "right"?

GMA ≠ CME

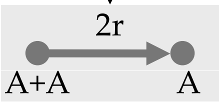


ODE
=
ODE
Continuous Chemistry
≠
Process Algebra
Discrete Chemistry
CTMC
=
CTMC

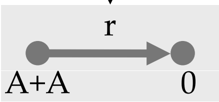
$A+A \rightarrow^{2r} A$ =? $A+A \rightarrow^{r} 0$

$[A]^\bullet = -2k[A]^2$ = $[A]^\bullet = -2k[A]^2$

$\boxed{k = r\gamma/2}$ $A+A \rightarrow^{2k} A$, $[A]_0=2/\gamma$	$A+A \rightarrow^{k} 0$, $[A]_0=2/\gamma$ $\boxed{k = r\gamma/2}$
$A+A \rightarrow^{2r} A$, $A+A$	$A+A \rightarrow^{r} 0$, $A+A$



≠



... as Automata

$[A]^\bullet = -2r\gamma[A]^2$ = $[A]^\bullet = -4k[A]^2$ = $[A]^\bullet = -4k[A]^2$ = $[A]^\bullet = -2r\gamma[A]^2$

$\boxed{k = r\gamma/2}$ $A+A \to^{4k} A$ $[A]_0=2/\gamma$

$A+A \to^{2k} 0$ $[A]_0=2/\gamma$ $\boxed{k = r\gamma/2}$

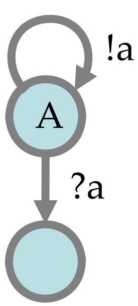
$A = ?a_{(2r)};0 \oplus !a_{(2r)};A$
$A \,|\, A$

$A+A \to^{4r} A$
$A+A$

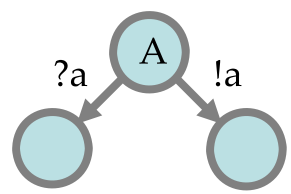
$A+A \to^{2r} 0$
$A+A$

$A = ?a_{(r)};0 \oplus !a_{(r)};0$
$A \,|\, A$

$A|A \xrightarrow{4r} A$ = $A+A \xrightarrow{4r} A$ ≠ $A+A \xrightarrow{2r} 0$ = $A|A \xrightarrow{2r} 0$



(a@2r)



(a@r)

Conclusions

Conclusions

http://LucaCardelli.name

- Devising 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.
- Parametric Processes (not shown)
  - An standard extension for the modular description of components.
- PolyAutomata (not shown)
  - Artificial ***Bio***-Chemistry: complexation and polymerization.
- An "artificial biochemistry"
  - A scalable mathematical and computational modeling framework.
  - To investigate "real biochemistry" on a large scale.

http://LucaCardelli.name

Q?