Artificial Biochemistry

Luca Cardelli

Microsoft Research

Longo Symposium
Paris, 2007-06-29

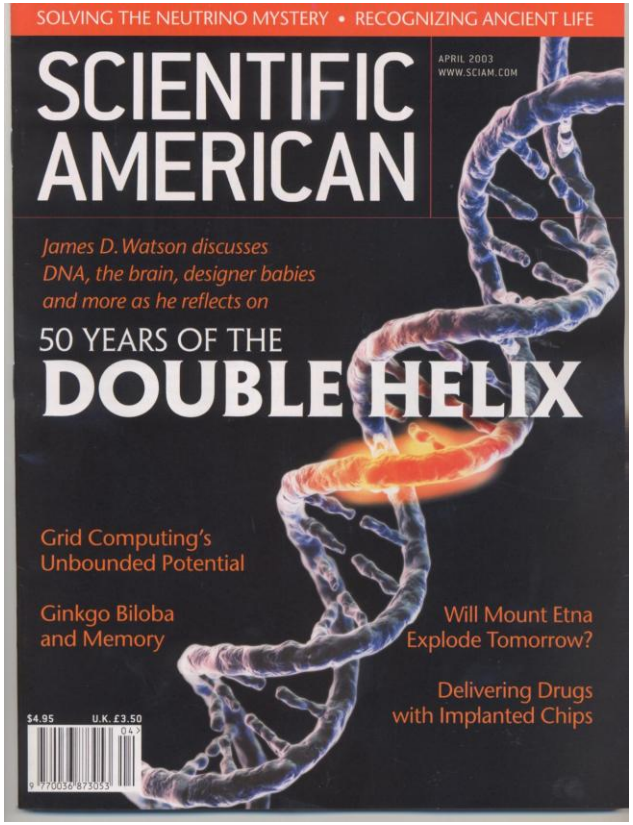
http://LucaCardelli.name

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

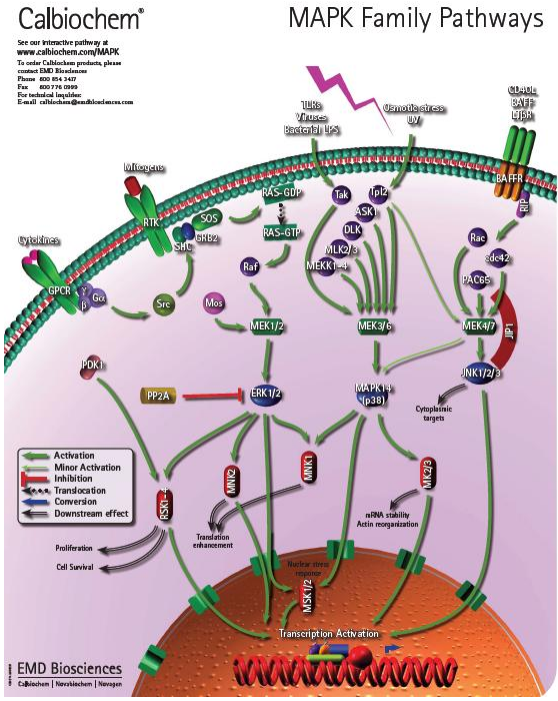
- Proteins (~1M coded from ~30K genes) are made of amino acids strings
  - Catalyze all biochemical reactions
  - Control metabolism (energy & materials)
  - Process signals, activate genes

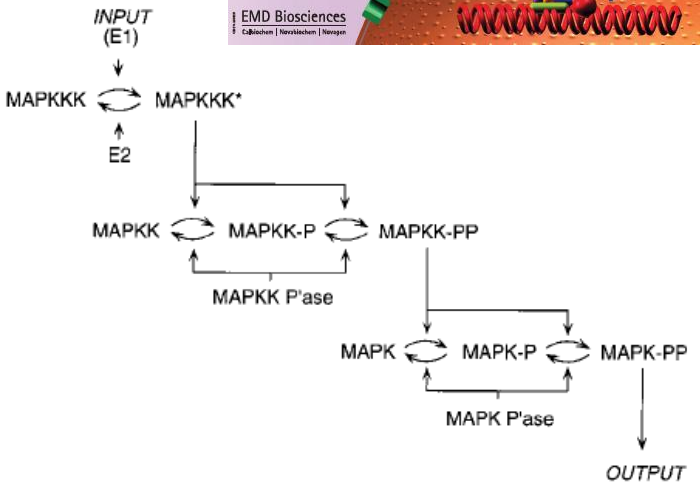
- Bootstrapping still a mystery
  - DNA, RNA, proteins, membranes are today interdependent. Not clear who came first
  - Not understood, not essential for us



Cells Compute

- No life without computation!
  - Finding food
  - Avoiding predators
- 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
  - 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.



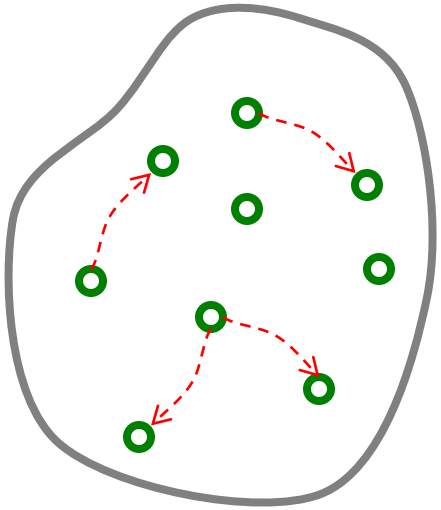


Ultrasensitivity in the mitogen-activated protein **cascade**, Chi-Ying F. Huang and James E. Ferrell, Jr., 1996, *Proc. Natl. Acad. Sci. USA*, 93, 10078-10083.

Stochastic Collectives

Stochastic Collectives

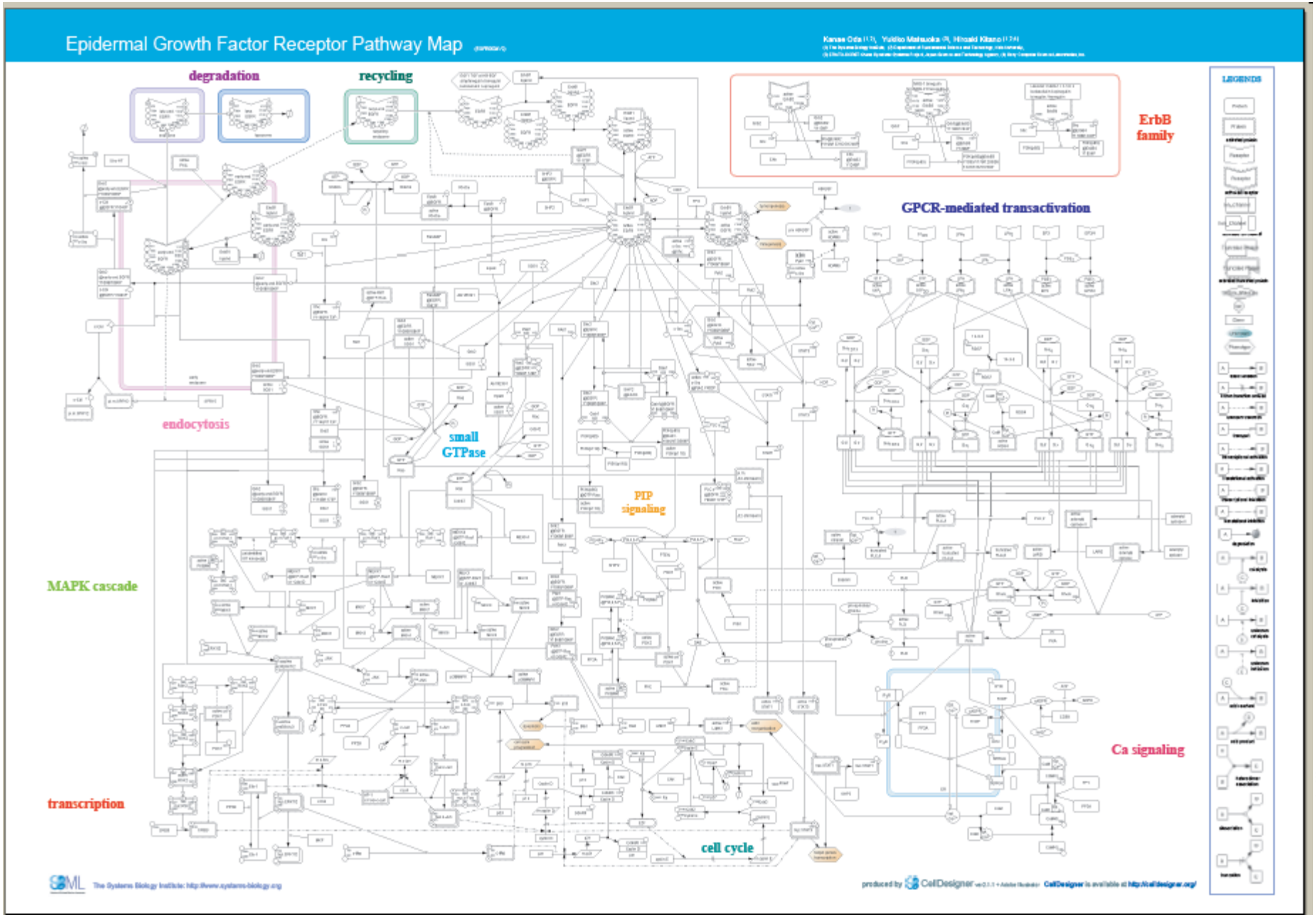
- "Collective":
  - A large set of interacting finite state automata:
    - Not quite language automata ("large set")
    - Not quite cellular automata ("interacting" but not on a grid)
    - Not quite process algebra ("collective behavior")
    - Cf. multi-agent systems and swarm intelligence



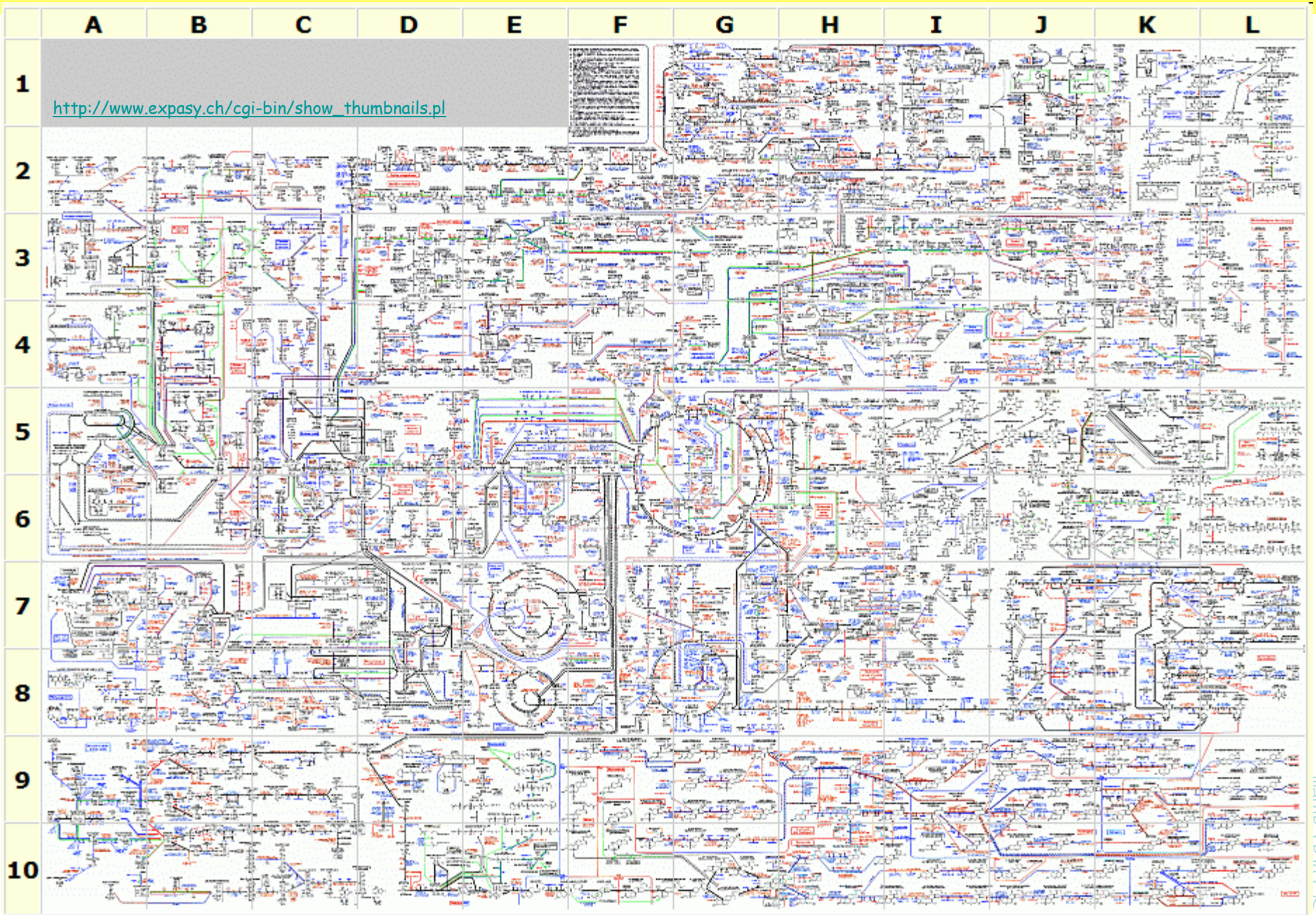
- "Stochastic":
  - Interactions have *rates*
    - Not quite discrete (hundreds or thousands of components)
    - Not quite continuous (non-trivial stochastic effects)
    - Not quite hybrid (no "switching" between regimes)

- 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].

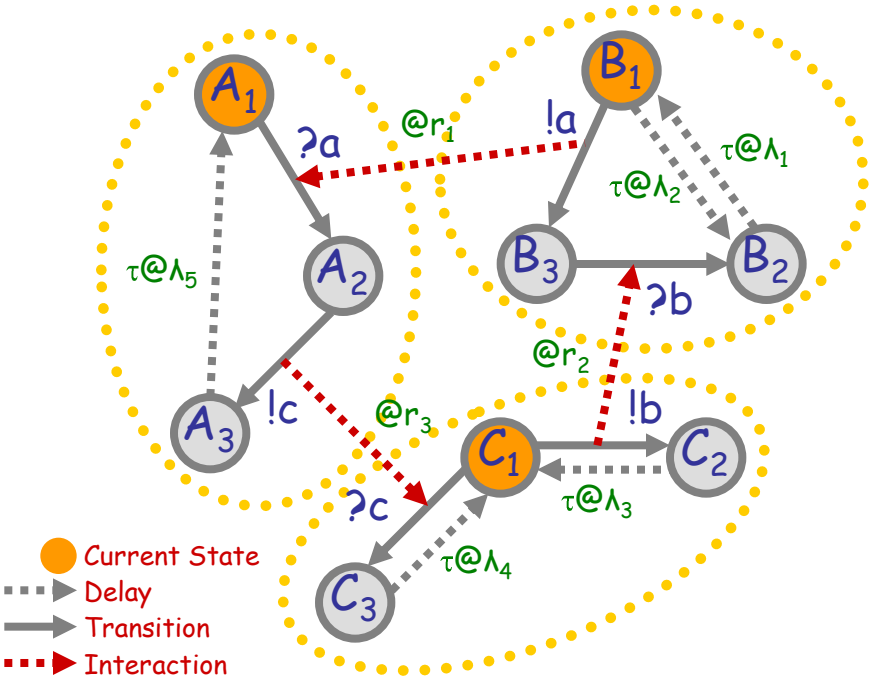
State Transitions



Compositionality (NOT!)



Interacting Automata



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]

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

Communication channels

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

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

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

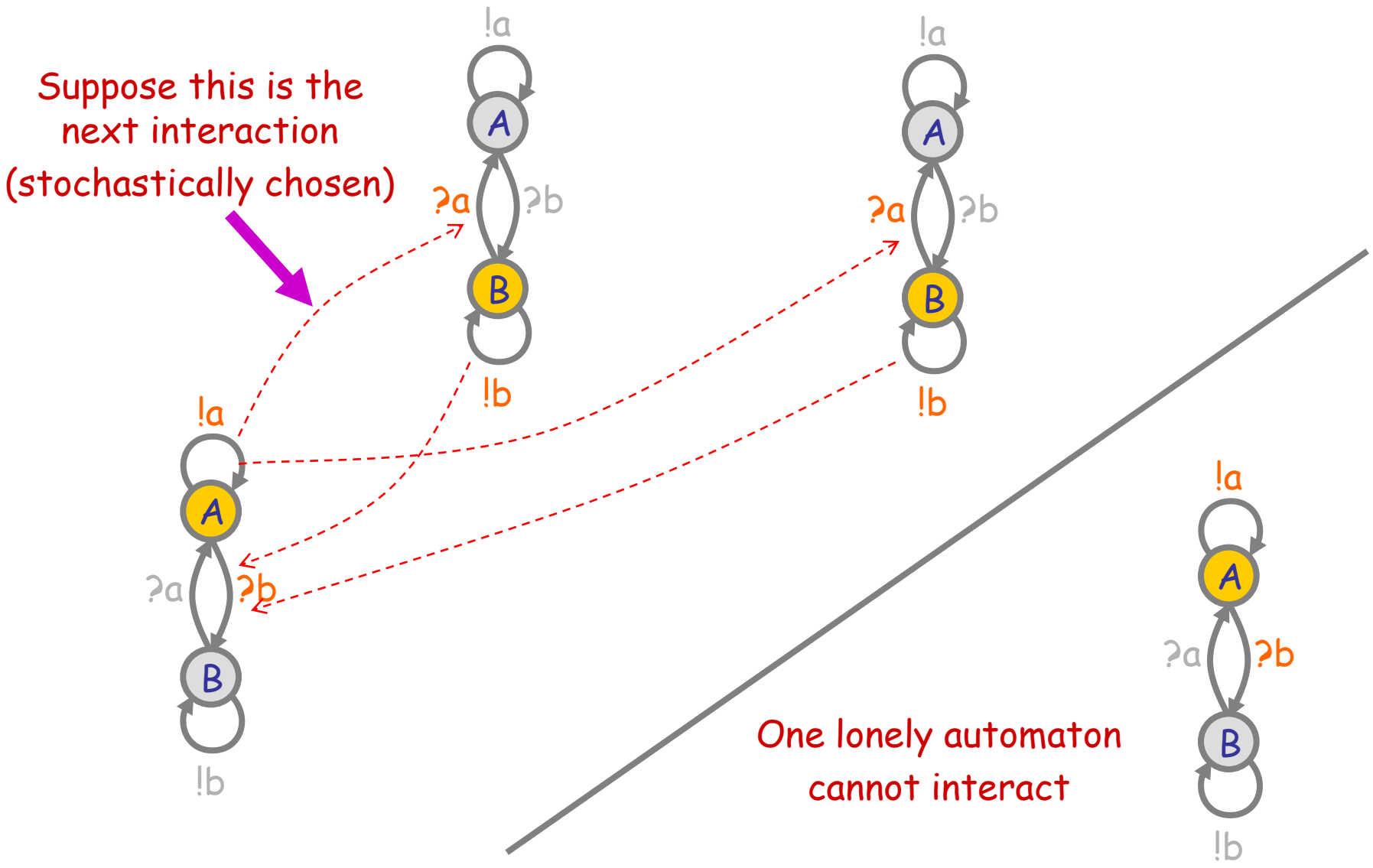
Automata

$A_1 \mid B_1 \mid C_1$

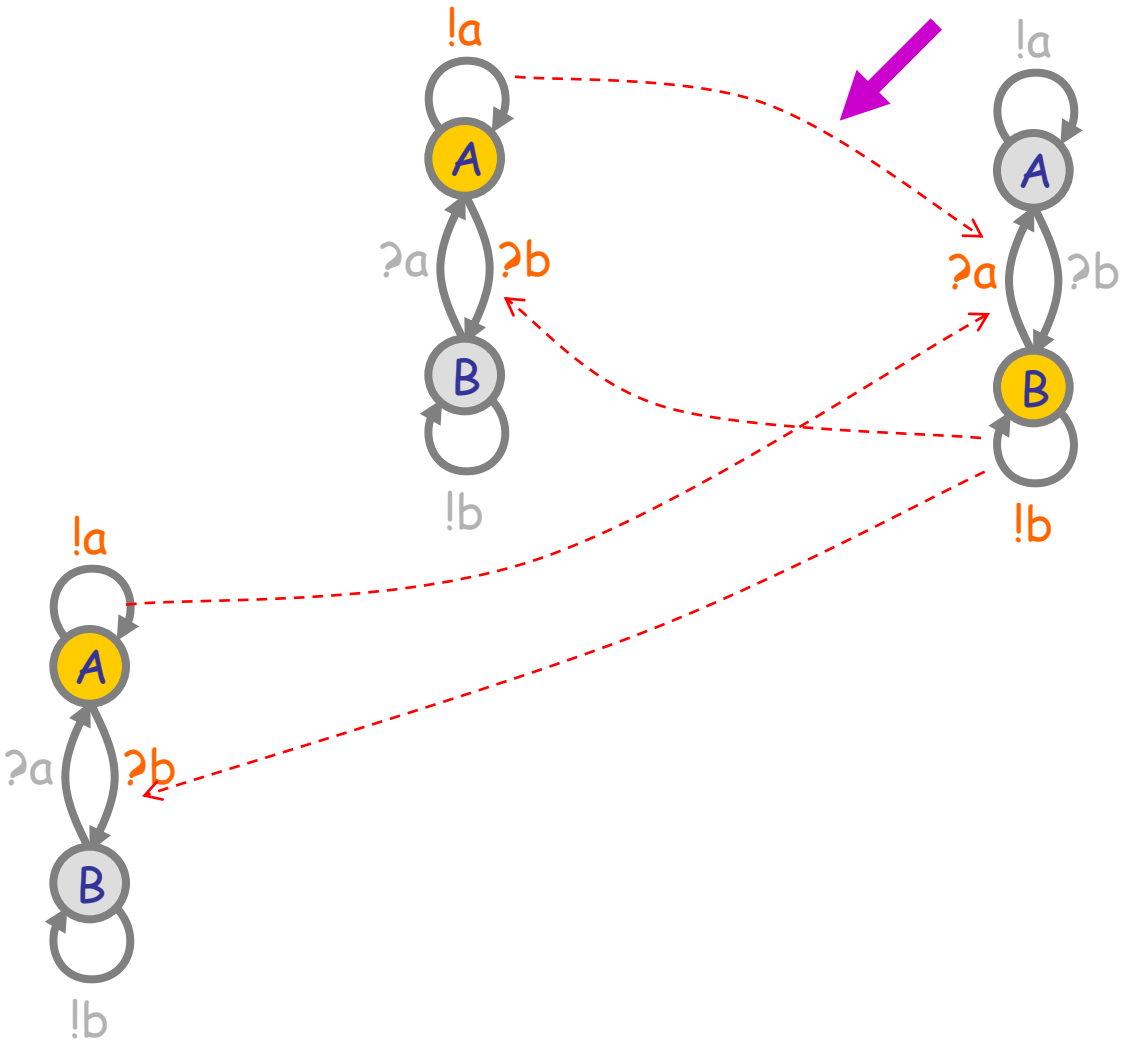
The system and initial state

Interactions have rates. Actions DO NOT have rates.

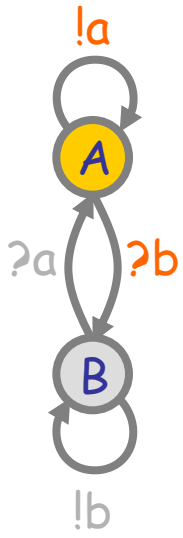
Interactions in a Population

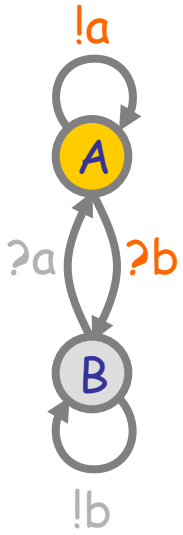


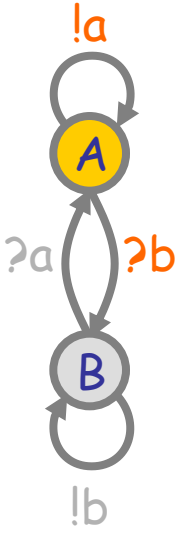
Interactions in a Population



Interactions in a Population

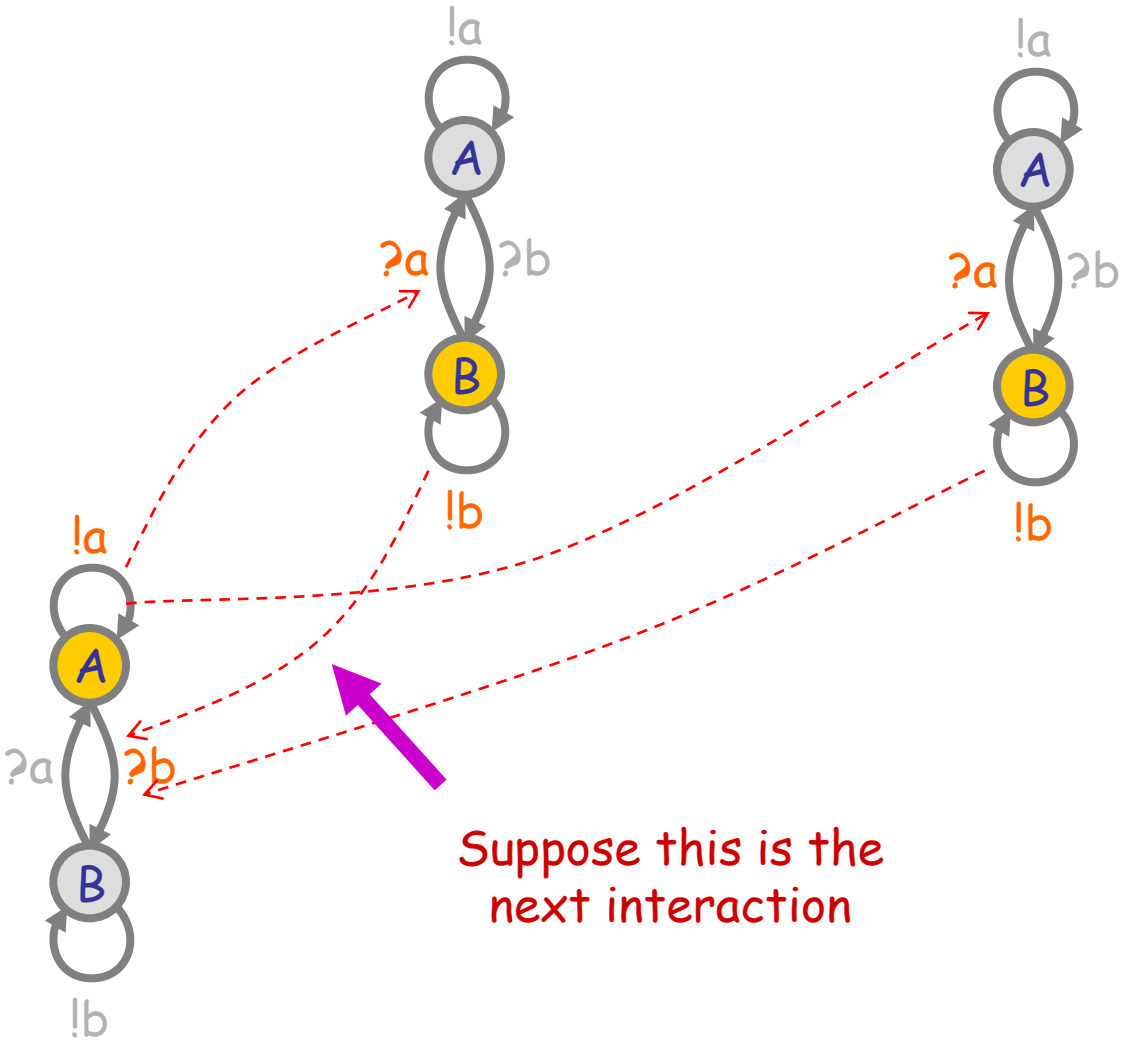




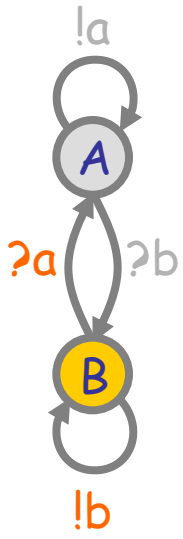


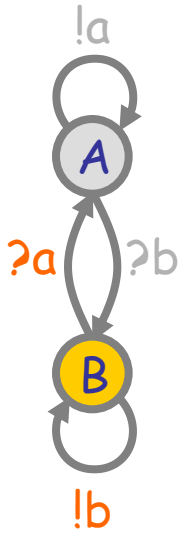
All-A stable population

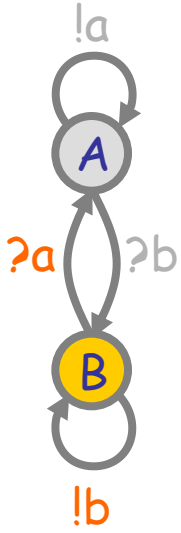
Interactions in a Population (2)



Interactions in a Population (2)



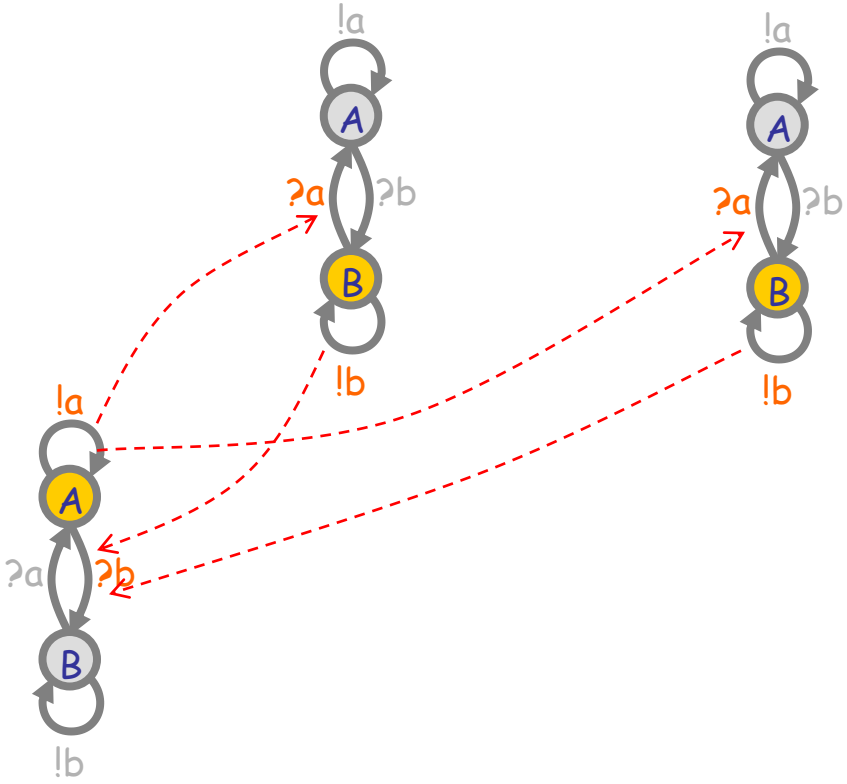


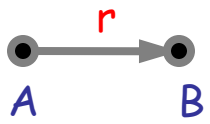


All-B stable population

Nondeterministic population behavior ("multistability")

CTMC Semantics



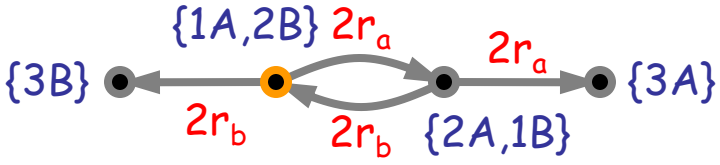


CTMC
(homogeneous) Continuous Time Markov Chain
- directed graph with no self loops
- nodes are system states
- arcs have transition rates

Probability of holding in state A:

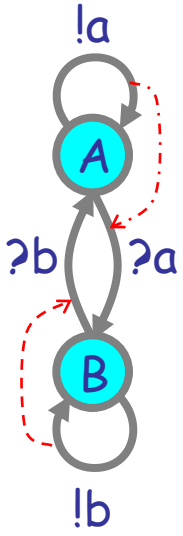
$Pr(H_A>t) = e^{-rt}$

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



CTMC

Groupies and Celebrities



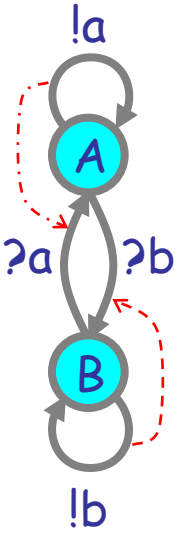
Celebrity

(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

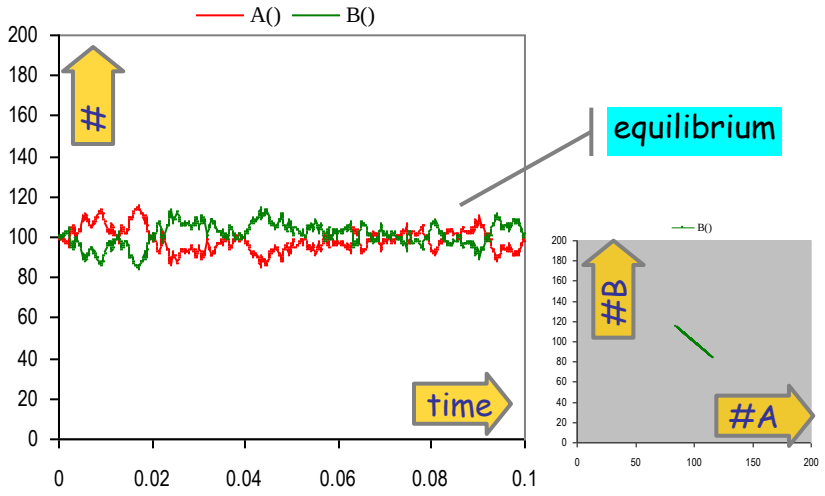
(wants to be like somebody different)

```
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

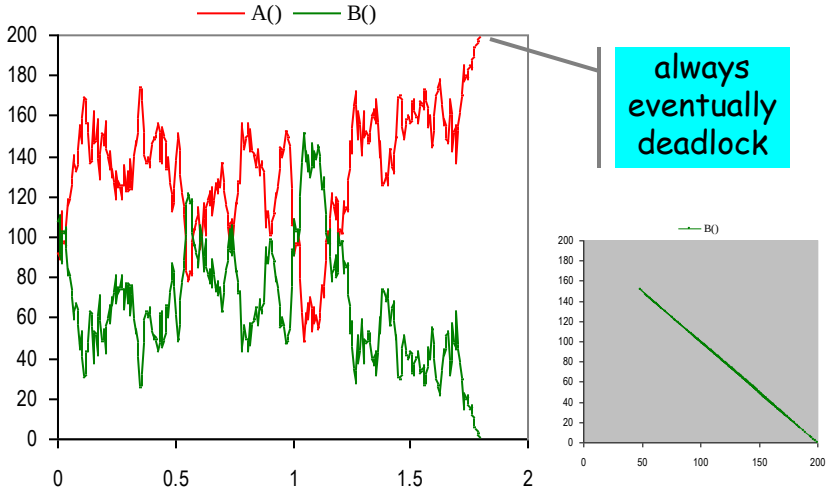
b@1.0

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:

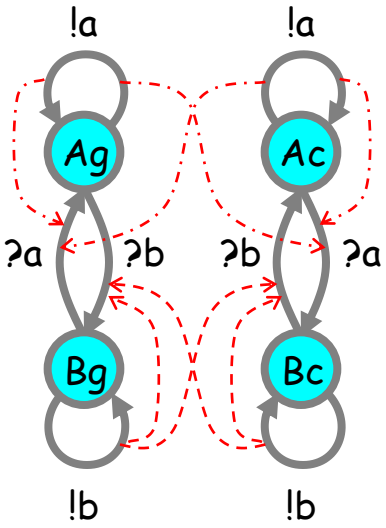


Unstable because within an A majority, an A has difficulty finding a B to emulate, but the few B's have plenty of A's to emulate, so the majority may switch to B. Leads to deadlock when everybody is in the same state and there is nobody different to emulate.

Both Together

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

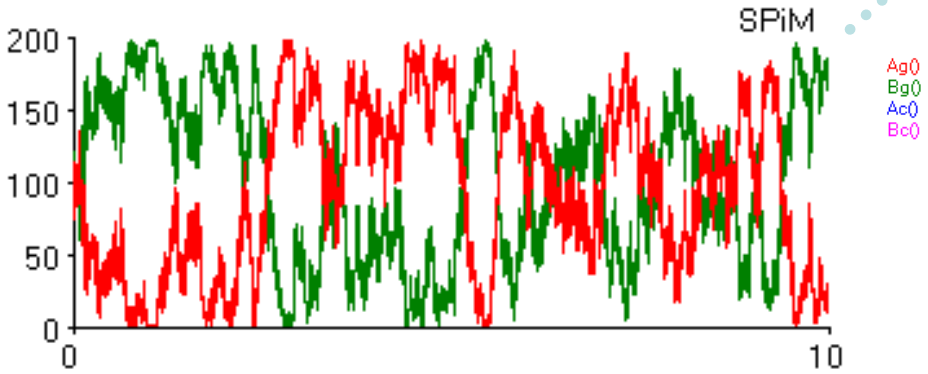
Many Groupies



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

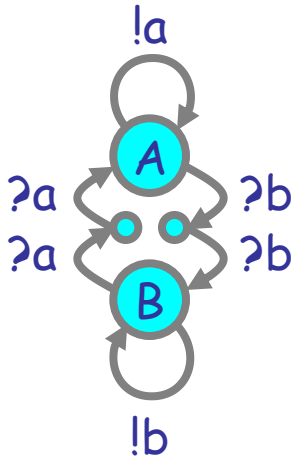


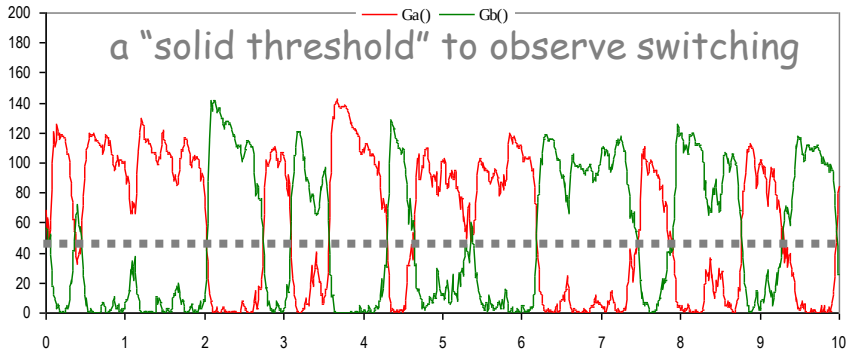
A tiny bit of "noise" can make a huge difference

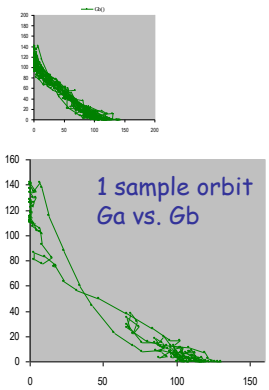
Regularity can arise not far from chaos

Hysteric Groupies

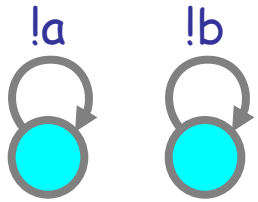
We can get more regular behavior from groupies if they "need more convincing", or "hysteresis" (history-dependence), to switch states.







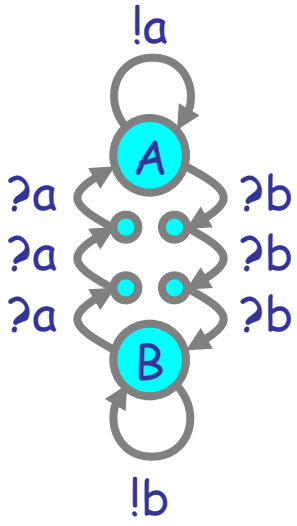

```
directive sample 10.0 1000
directive plot Ga(); Gb()

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())
```

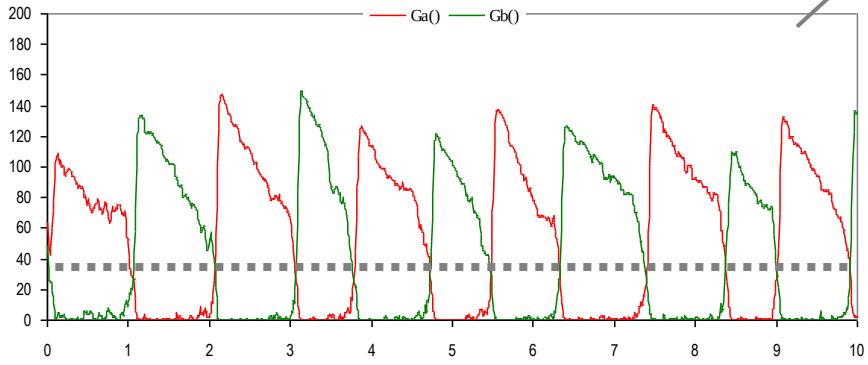


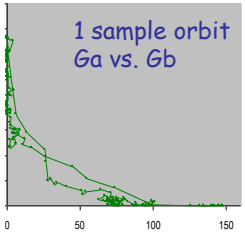
(With doping to break deadlocks)

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

"regular" oscillation



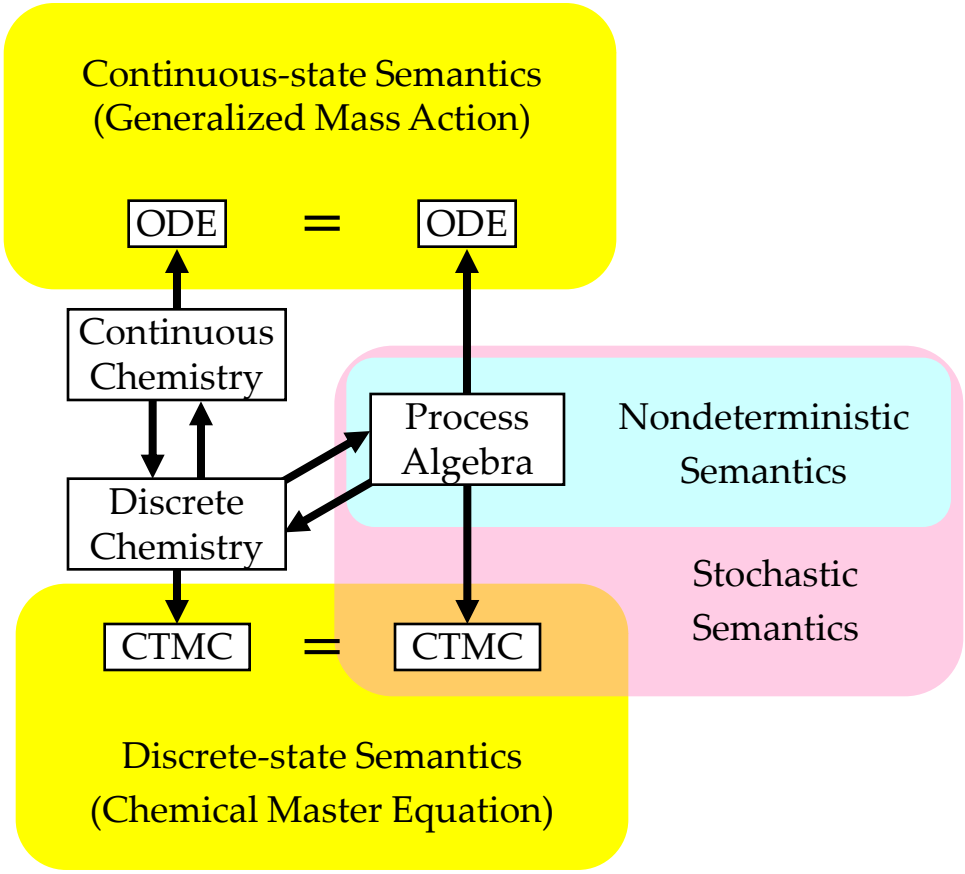





```
directive sample 10.0 1000
directive plot Ga(); Gb()

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)

$E ::= 0 \mid X=M, E$ Reagents

$M ::= 0 \mid \pi;P \oplus M$ Molecules

$P ::= 0 \mid X \mid P$ Solutions

$\pi ::= \tau_{(r)} \mid ?a_{(r)} \mid !a_{(r)}$ Interactions (delay, input, output)

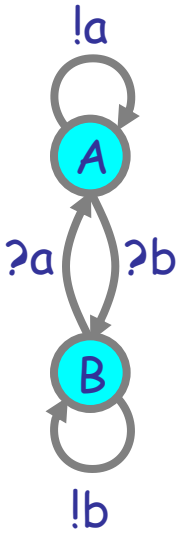
$CGF ::= E,P$ Reagents plus Initial Conditions

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 ($M \oplus 0 = 0 \oplus M = M$)
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

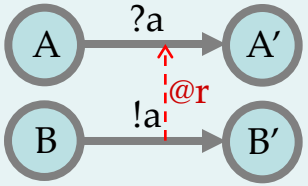
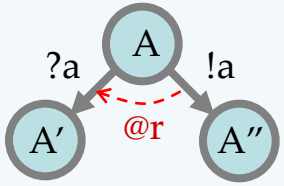
$B = !b;B \oplus ?a;A$ — Automaton in state B

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

Processes to Chemistry

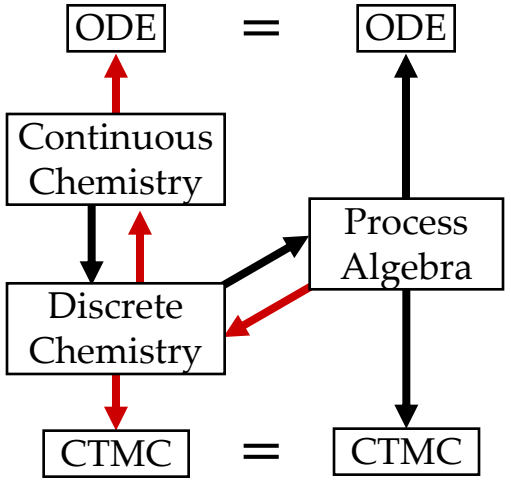
V = interaction volume

N_A = Avogadro's number

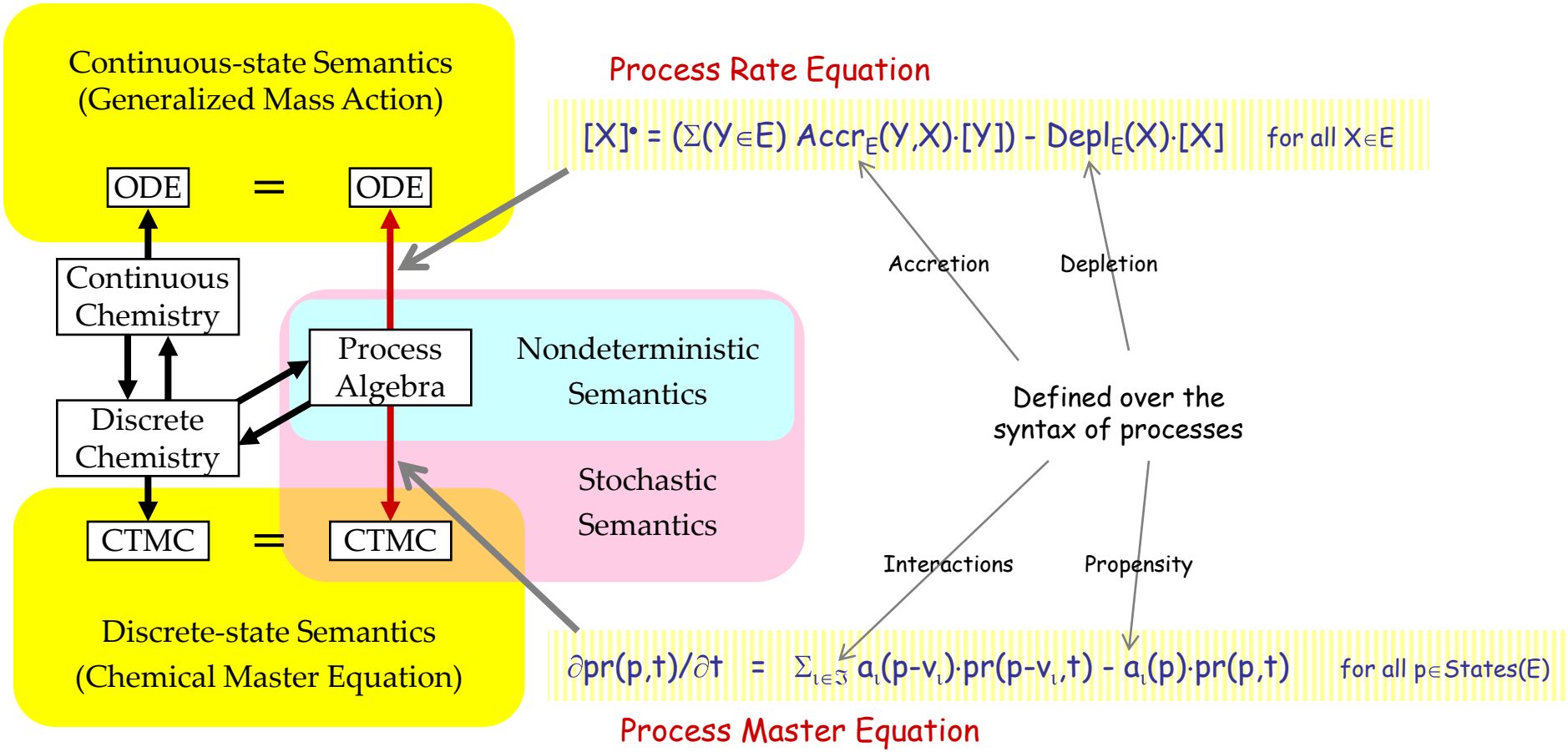
Automata →	Discrete Chemistry →	Continuous Chemistry	$\gamma = N_A V$
initial states A | A | ... | A	initial quantities $\#A_0$	initial concentrations $[A]_0$	with $[A]_0 = \#A_0/\gamma$
	$A \dashrightarrow^{r} A'$	$A \rightarrow^{k} A'$	with $k = r$
	$A+B \dashrightarrow^{r} A'+B'$	$A+B \rightarrow^{k}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

Think $\gamma = 1$

i.e. $V = 1/N_A$



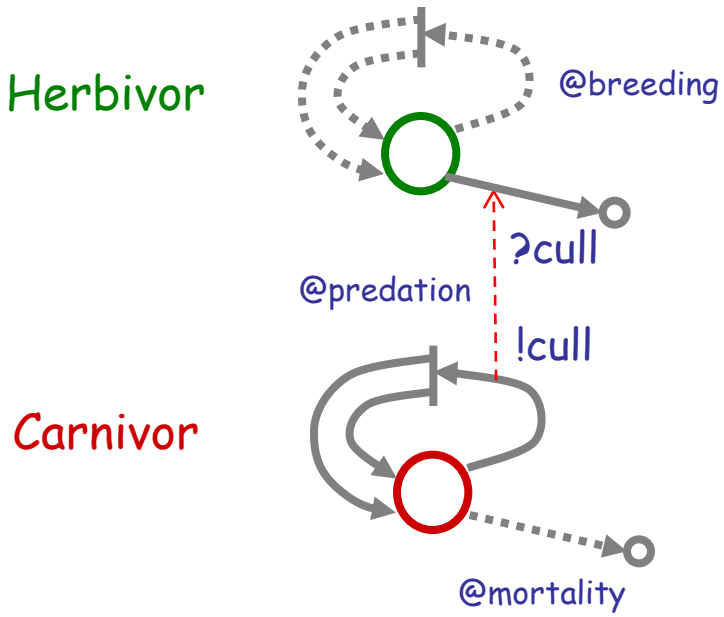
Quantitative Process Semantics



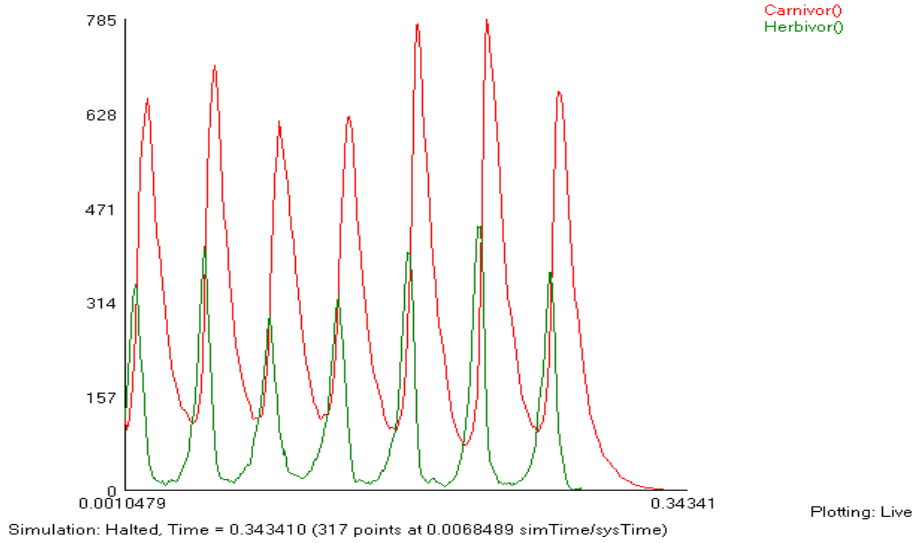
Lotka-Volterra

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

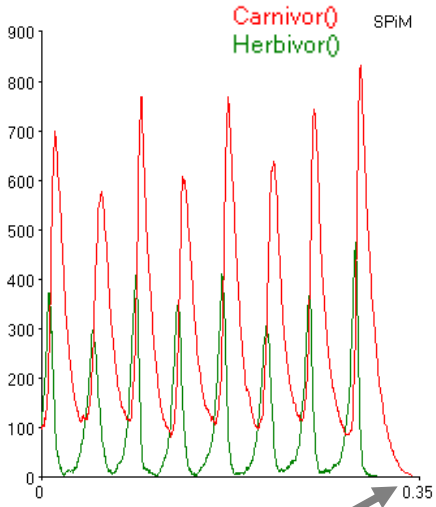
$H = \tau_b; (H|H) \oplus ?c_{(p)}; 0$
$C = \tau_m; 0 \oplus !c_{(p)}; (C|C)$
$\#H_0, \#C_0$

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

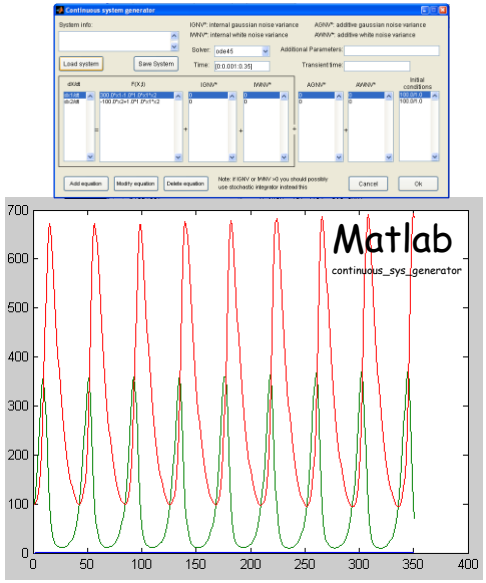
$[H]^{\bullet} = b[H] - p\gamma[H][C]$
$[C]^{\bullet} = -m[C] + p\gamma[H][C]$
$[H]_0 = \#H_0/\gamma$
$[C]_0 = \#C_0/\gamma$

m=100.0
b=300.0
p=1.0
γ=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()
```



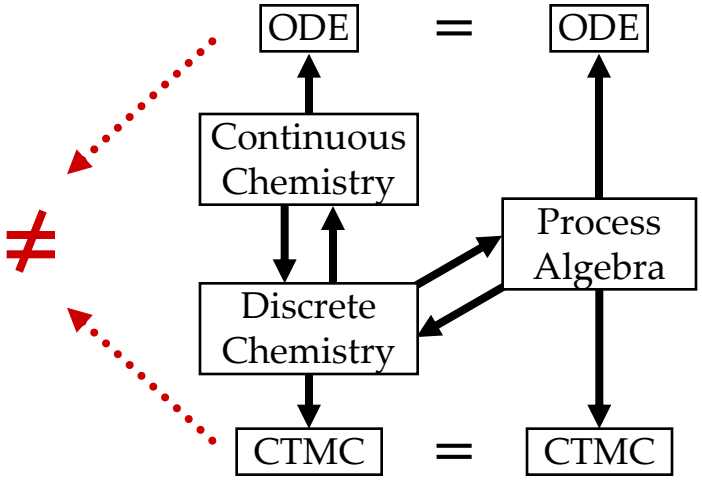
Extinction



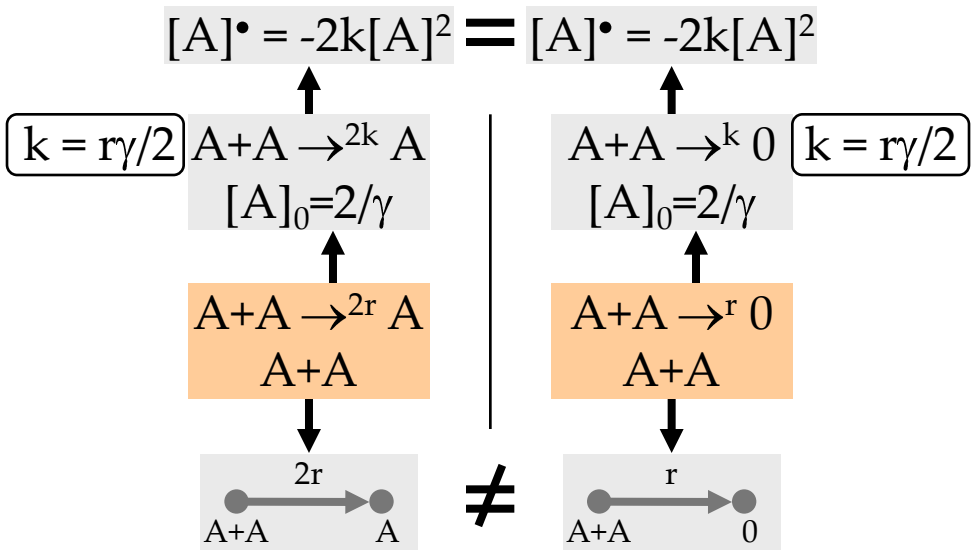
No extinction

Which one is "right"?

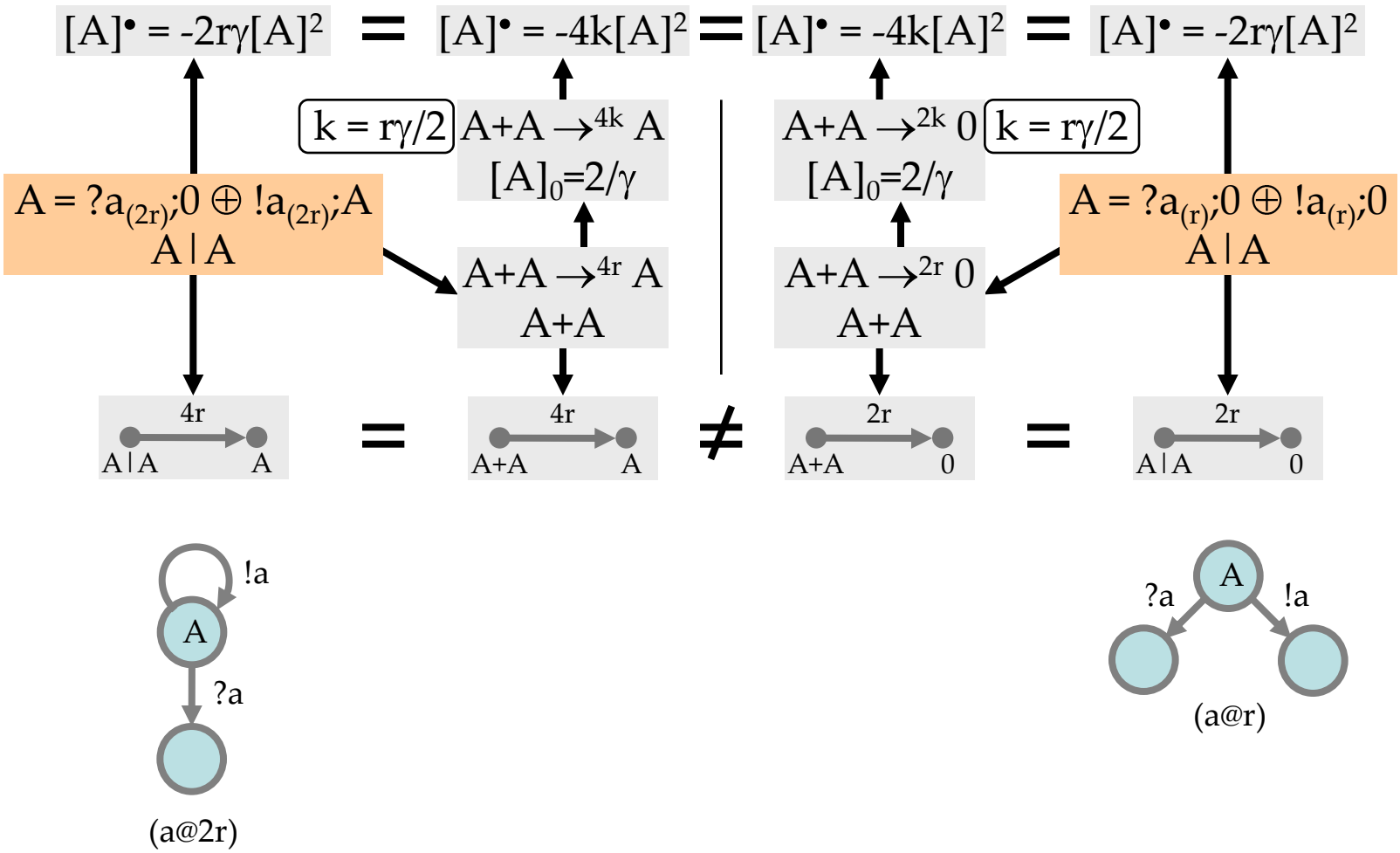
GMA ≠ CME


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

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



... as Automata


[A]• = -2rγ[A]²
=
[A]• = -4k[A]²
=
[A]• = -4k[A]²
=
[A]• = -2rγ[A]²
k = rγ/2
A+A →4k A
[A]0=2/γ
A+A →2k 0
[A]0=2/γ
k = rγ/2
A = ?a(2r);0 ⊕ !a(2r);A
A | A
A+A →4r A
A+A
A+A →2r 0
A+A
A = ?a(r);0 ⊕ !a(r);0
A | A
4r
A | A
A
=
4r
A+A
A
≠
2r
A+A
0
=
2r
A | A
0
!a
A
?a
(a@2r)
A
?a
!a
(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.