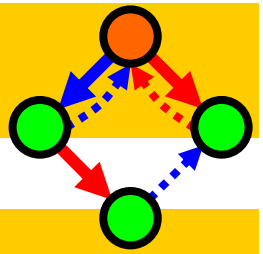
Research is what I'm doing when I don't know what I'm doing. Wernher Von Braun.

Artificial Biochemistry



Varieties of Stochastic Calculi

Luca Cardelli

Microsoft Research

The Microsoft Research - University of Trento
Centre for Computational and Systems Biology

Trento, 2006-05-22..26

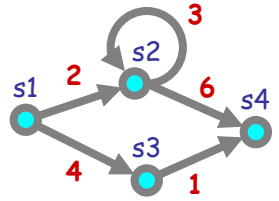
www.luca.demon.co.uk/ArtificialBiochemistry.htm

Continuous Time Markov Chains

A continuous-time Markov chain (CTMC) is a pair $M=(S,R)$ where S is a countable set of *states*, and $R: S \times S \to \mathbf{Real}_{\geq 0}$ is the *rate matrix*(1).

R	s1	s2	s3	s4
s1	0	2	4	0
s2	0	3	0	6
s3	0	0	0	1
s4	0	0	0	0

=



The exit rate $E(s) = \Sigma(s' \in S)\, R(s,s')$ denotes that the probability of taking a transition from s within t time units equals $1-e^{-E(s)t}$.
(The time between state changes is a random variable with a memoryless distribution.)

If $R(s,s')>0$ for more than one state s', then there is a *race* between the outgoing transitions of s.

E(s1) = 6
E(s2) = 9
E(s3) = 1
E(s4) = 0

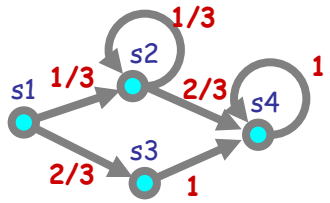
The embedded discrete-time Markov chain.
(N.B. when we do this, we loose information on holding time in each state)
The probability $P(s,s')$ of moving from s to s' in a single step is:

if $E(s)>0$ then $P(s,s')=R(s,s')/E(s)$

if $E(s)=0$ then $P(s,s)=1$ and $P(s,s')=0$ for $s \neq s'$.

P	s1	s2	s3	s4
s1	0	1/3	2/3	0
s2	0	1/3	0	2/3
s3	0	0	0	1
s4	0	0	0	1

=



$P(s,s')$ equals the probability that the delay of going from s to s' "finishes before" the delay of any other outgoing transition from s.

(1) The more standard *generator matrix* has $-E(s)$ on the diagonal, which requires forbidding self-loops.

Model checking continuous-time Markov chains by transient analysis

Christel Baier[a], Boudewijn Haverkort[b], Holger Hermanns[c] and Joost-Pieter Katoen[c,*]

CTMC Modeling Formalism

- CTMCs
  - Finite State Machines with rates on transitions.
  - All durations are exponentially distributed (memoryless)
  - Well studied class of stochastic processes
  - Efficient analysis algorithms for stationary and transient analysis

- We consider only:
  - Modeling formalisms that map to CTMC's
  - But there are still quite a few of those:

- High level formalisms mapping to CTMCs
  - Stochastic Petri Nets [Molloy]
  - Markovian Queuing Networks [Muppala & Triverdi]
  - Stochastic Automata Networks [Plateau]
  - Probabilistic I/O Automata [Wu et al.]
  - Stochastic Process Algebras [Herzog et al.] [Hillston]

Stochastic Calculi

PEPA

Labeled Transition Semantics

$$P ::= (\alpha, r).P \mid P + Q \mid P \underset{L}{\bowtie} Q \mid P/L \mid A$$

A Compositional Approach to Performance Modelling

http://www.dcs.ed.ac.uk/pepa/book.pdf

Jane Hillston

Definition 3.3.1 *The* apparent rate *of action of type α in a component P, denoted $r_\alpha(P)$ is the sum of the rates of all activities of type α in $\mathcal{A}ct(P)$.*

1. $r_\alpha((\beta, r).P) = \begin{cases} r & \text{if } \beta = \alpha \\ 0 & \text{if } \beta \neq \alpha \end{cases}$
2. $r_\alpha(P + Q) = r_\alpha(P) + r_\alpha(Q)$
3. $r_\alpha(P/L) = \begin{cases} r_\alpha(P) & \text{if } \alpha \notin L \\ 0 & \text{if } \alpha \in L \end{cases}$
4. $r_\alpha(P \underset{L}{\bowtie} Q) = \begin{cases} \min(r_\alpha(P), r_\alpha(Q)) & \text{if } \alpha \in L \\ r_\alpha(P) + r_\alpha(Q) & \text{if } \alpha \notin L \end{cases}$

Prefix

$$\frac{}{(\alpha, r).E \xrightarrow{(\alpha,r)} E}$$

Choice

$$\frac{E \xrightarrow{(\alpha,r)} E'}{E + F \xrightarrow{(\alpha,r)} E'} \qquad \frac{F \xrightarrow{(\alpha,r)} F'}{E + F \xrightarrow{(\alpha,r)} F'}$$

Cooperation

$$\frac{E \xrightarrow{(\alpha,r)} E'}{E \underset{L}{\bowtie} F \xrightarrow{(\alpha,r)} E' \underset{L}{\bowtie} F} (\alpha \notin L) \qquad \frac{F \xrightarrow{(\alpha,r)} F'}{E \underset{L}{\bowtie} F \xrightarrow{(\alpha,r)} E \underset{L}{\bowtie} F'} (\alpha \notin L)$$

$$\frac{E \xrightarrow{(\alpha,r_1)} E' \quad F \xrightarrow{(\alpha,r_2)} F'}{E \underset{L}{\bowtie} F \xrightarrow{(\alpha,R)} E' \underset{L}{\bowtie} F'} (\alpha \in L) \quad \text{where } R = \frac{r_1}{r_\alpha(E)} \frac{r_2}{r_\alpha(F)} \min(r_\alpha(E), r_\alpha(F))$$

Hiding

$$\frac{E \xrightarrow{(\alpha,r)} E'}{E/L \xrightarrow{(\alpha,r)} E'/L} (\alpha \notin L) \qquad \frac{E \xrightarrow{(\alpha,r)} E'}{E/L \xrightarrow{(\tau,r)} E'/L} (\alpha \in L)$$

Constant

$$\frac{E \xrightarrow{(\alpha,r)} E'}{A \xrightarrow{(\alpha,r)} E'} (A \stackrel{def}{=} E)$$

(ordinary) π-calculus

Reduction Semantics
(with structural congruence)

Syntax

$$\begin{array}{rcll} \pi & ::= & x(y) & \text{receive } y \text{ along } x \\ & & \overline{x}\langle y\rangle & \text{send } y \text{ along } x \end{array}$$

$$P \;::=\; 0 \mid \textstyle\sum_{i\in I} \pi_i.P_i \mid [x=y]\,P \mid P_1|P_2 \mid (\text{new } x)P \mid !P$$

Structural congruence

Renaming of bound variables

$$\begin{array}{rcll} x(y).P & = & x(z).(\{z/y\}\,P) & \text{if } z \notin FN(P) \\ (\text{new } y).P & = & (\text{new } z).(\{z/y\}\,P) & \text{if } z \notin FN(P) \end{array}$$

Structural congruence laws

$$\begin{array}{rcll} P|Q & \equiv & Q|P & \text{commutativity of parallel composition} \\ (P|Q)|R & \equiv & P|(Q|R) & \text{associativity of parallel composition} \\ P+Q & \equiv & Q+P & \text{commutativity of summation} \\ (P+Q)+R & \equiv & P+(Q+R) & \text{associativity of summation} \\ (\text{new } x)0 & \equiv & 0 & \text{restriction of inert processes} \\ (\text{new } x)(\text{new } y)P & \equiv & (\text{new } y)(\text{new } x)P & \text{polyadic restriction} \\ ((\text{new } x)P)|Q & \equiv & (\text{new } x)(P|Q) \quad \text{if } x \notin FN(Q) & \text{scope extrusion} \\ !P & \equiv & P|!P & \text{replication} \end{array}$$

Reaction rules

$$(\cdots + \overline{x}\langle z\rangle.Q)|(\cdots + x(y).P) \to Q|P\{z/y\} \qquad \text{communication (COMM)}$$

$$\frac{P \to P'}{P|Q \to P'|Q} \qquad \text{reaction under parallel composition (PAR)}$$

$$\frac{P \to P'}{(\text{new } x)P \to (\text{new } x)P'} \qquad \text{reaction under restriction (RES)}$$

$$\frac{Q \equiv P \;\; P \to P' \;\; P' \equiv Q'}{Q \to Q'} \qquad \text{structural congruence (STRUCT)}$$

Syntax

Chemical Mixing

Reactions

Stochastic π-calculus

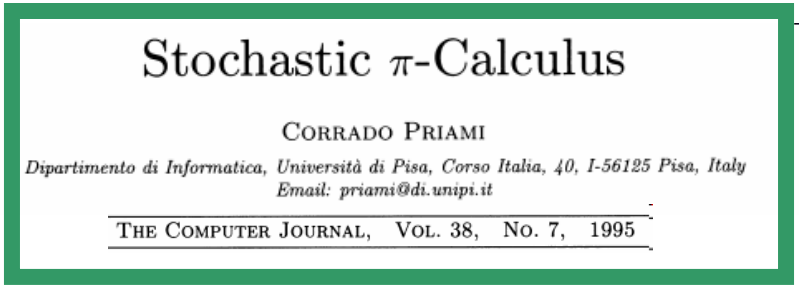
Labeled Transition Semantics

- Stochastic extension of π-calculus.
  - Associate a single parameter r (rate) in (0, infinity] to each activity a.
  - The rate and the associated exponential distribution describes the stochastic behavior of the activity.

a.P is replaced by a@r.P

- Exponential distribution
  - Guarantees the memoryless property: the time at which a change of state occurs is independent of the time at which the last change of state occurred.
  - I.e. we do not have to remember past states transitions (e.g. in the implementation).
  - Conversely, if we *do* want to remember past transitions, we can do it by programming.

- Race resolution
  - Races are resolved in a probabilistic competitive way: all the activities that are enabled in a state compete and the fastest one (stochastically) succeeds. [Gillespie]



Stochastic π-Calculus

CORRADO PRIAMI

Dipartimento di Informatica, Università di Pisa, Corso Italia, 40, I-56125 Pisa, Italy
Email: priami@di.unipi.it

THE COMPUTER JOURNAL, VOL. 38, NO. 7, 1995

$$Act : (\mu,r).P \xrightarrow{(\mu,r)} P,\ \mu \text{ not input} \qquad Ein : (x(y),r).P \xrightarrow{(xw,r)} P[w/y] \qquad Ide : \frac{P\{\tilde{y}/\tilde{x}\} \xrightarrow{\theta} P'}{Q(\tilde{y}) \xrightarrow{\theta} P'}, Q(\tilde{x}) = P$$

$$Par_0 : \frac{P \xrightarrow{\theta} P'}{P|Q \xrightarrow{||_0\theta} P'|Q}, bn(\ell(\theta)) \cap fn(Q) = \emptyset \qquad Sum_0 : \frac{P \xrightarrow{\theta} P'}{P+Q \xrightarrow{+_0\theta} P'} \qquad Res : \frac{P \xrightarrow{\theta} P'}{(\nu x)P \xrightarrow{\theta} (\nu x)P'}, x \notin n(\ell(\theta))$$

$$Par_1 : \frac{P \xrightarrow{\theta} P'}{Q|P \xrightarrow{||_1\theta} Q|P'}, bn(\ell(\theta)) \cap fn(Q) = \emptyset \qquad Sum_1 : \frac{P \xrightarrow{\theta} P'}{Q+P \xrightarrow{+_1\theta} P'} \qquad Open : \frac{P \xrightarrow{\vartheta(\tilde{x}y,r)} P'}{(\nu y)P \xrightarrow{\vartheta(\tilde{x}(y),r)} P'}, y \neq x$$

$$Com_0 : \frac{P \xrightarrow{\vartheta(\tilde{x}y,r_p)} P', Q \xrightarrow{\vartheta'(\tilde{x}y,r_q)} Q'}{P|Q \xrightarrow{(\langle||_0\vartheta\tilde{x}y,||_1\vartheta'xy\rangle,R(P,\tilde{x}y,Q,xy,r_p,r_q))} P'|Q'}$$

$$Close_0 : \frac{P \xrightarrow{\vartheta(\tilde{x}(y),r_p)} P', Q \xrightarrow{\vartheta'(xy,r_q)} Q'}{P|Q \xrightarrow{(\langle||_0\vartheta\tilde{x}(y),||_1\vartheta'xy\rangle,R(P,\tilde{x}(y),Q,xy,r_p,r_q))} (\nu y)(P'|Q')}, y \notin fn(Q)$$

TABLE 2. Early proved transition system of $S\pi$.

The (Biochemical) Sπ

Labeled Transition Semantics

Application of a stochastic name-passing calculus to representation and simulation of molecular processes

Corrado Priami [a,*], Aviv Regev [b,c], Ehud Shapiro [b,d], William Silverman [b]

Information Processing Letters 80 (2001) 25–31

This paper was motivated by the need to model "chemical" interaction laws and mixed choice. But it supports mixed choice, only over a distinguished set of channels with exactly 1 mixed choice per summation (homodimerization).

Table 1
Reduction semantics of the biochemical stochastic π-calculus

$$\left(\cdots + (\overline{x}\langle z\rangle, r).Q\right) \Big| \left((x(y), r).P + \cdots\right) \xrightarrow{x, r_b\cdot 1\cdot 1} Q|P\{z/y\}, \quad x \notin \mathcal{H},$$

$$\left(\cdots + (\overline{x}\langle z\rangle, r).Q + (x(y), r).P\right) \Big|$$
$$\left((\overline{x}\langle z\rangle, r).Q + (x(y), r).P + \cdots\right) \xrightarrow{x, 1/2\cdot r_b\cdot 2\cdot(2-1)} Q|P\{z/y\}, \quad x \in \mathcal{H},$$

$$\frac{P \xrightarrow{x, r_b\cdot r_0\cdot r_1} P'}{P|Q \xrightarrow{x, r_b\cdot r_0'\cdot r_1'} P'|Q}, \qquad \begin{cases} r_0' = r_0 + In_x(Q), \\ r_1' = r_1 + Out_x(Q), \end{cases}$$

$$\frac{P \xrightarrow{x, r_b\cdot r_0\cdot r_1} P'}{(\nu x)P \xrightarrow{x, r_b\cdot r_0\cdot r_1} (\nu x)P'} \qquad \frac{Q \equiv P,\ P \xrightarrow{x, r_b\cdot r_0\cdot r_1} P',\ P' \equiv Q'}{Q \xrightarrow{x, r_b\cdot r_0\cdot r_1} Q'}$$

$$In_x(\mathbf{0}) = 0,$$
$$In_x\left(\sum_{i\in I}(\pi_i, r_i).P_i\right) = \left|\left\{(\pi_i, r_i) \mid i \in I \wedge sbj(\pi_i) = x\right\}\right|,$$
$$In_x(P_1|P_2) = In_x(P_1) + In_x(P_2),$$
$$In_x\left((\nu z)P\right) = \begin{cases} In_x(P) & \text{if } z \neq x, \\ 0 & \text{otherwise.} \end{cases}$$

Out_x is similarly defined, by replacing any occurrence of In with Out and the condition $sbj(\pi_i) = x$ with $sbj(\pi_i) = \overline{x}$.

A usual reaction is implemented by the three parameters r_b, r_0 and r_1, where r_b represents the basal rate, and r_0 and r_1 denote the quantities of interacting molecules, and are computed compositionally via In_x and Out_x while deducing transitions. The first axiom in Table 1 corresponds to usual reactions, with two different molecules. The second one corresponds to homodimerization reactions.

StoPi

Reduction Semantics
(with structural congruence)

Definition 2.1 (Process Expressions)
The set of action prefixes π are defined by

$$\begin{array}{rll} \pi ::= & x(\vec{y}) & \text{receive } \vec{y} \text{ along } x \\ \mid & \overline{x}\langle\vec{y}\rangle & \text{send } \vec{y} \text{ along } x. \end{array}$$

where $\vec{y}$ denotes a tuple $(y_1, \ldots, y_n)$ of channel names called a channel vector. The set of process expressions in the StoPi-calculus is defined by the following syntax with the syntactic categories processes (P), guarded sums (M) and process declarations (D):

$$\begin{array}{rl} D ::= & D;D \mid A(\vec{x}) \stackrel{def}{=} P \\ P ::= & P|P \mid \text{new } (x_1 : r_1, \ldots, x_n : r_n)\ P \\ & \mid M \mid A(\vec{x}) \mid 0 \\ M ::= & \pi.P \mid M + M \end{array}$$

where 0 is the nil process (or empty sum) which cannot do any transitions and $A(\vec{x})$ is an invocation of the process with identifier A with actual parameters $\vec{x}$.

Definition 2.2 (Structural Congruence)
Two processes P and Q in the StoPi-calculus are structurally congruent, written $P \equiv Q$, if we can transform one into the other by using the following equations (in either direction):

1. $P \equiv \{\vec{x}/\vec{y}\}P$ *if* $y_i \notin fn(P)$ *for* $1 \leq i \leq |\vec{y}|$. *(α-conversion)*
2. $M+N \equiv N+M$, $(M+N)+L \equiv M+(N+L)$
3. $P|0 \equiv P$, $P|Q \equiv Q|P$, $P|(Q|R) \equiv (P|Q)|R$
4. new $\vec{x}\,0 \equiv 0$, new $\vec{x}\vec{y}\,P \equiv$ new $\vec{y}\vec{x}\,P$
   new $\vec{x}\,(P|Q) \equiv P|$new $\vec{x}\ Q$ *if* $x_i \notin fn(P)$
   for $1 \leq i \leq |\vec{x}|$,
5. $A(\vec{x}) \stackrel{def}{=} P \Rightarrow A(\vec{y}) \equiv \{\vec{y}/\vec{x}\}P$

$$[STRUCT] \quad \frac{P \xrightarrow{x} P'}{Q \xrightarrow{x} Q'} \quad \text{if } P \equiv Q \text{ and } P' \equiv Q'$$

$$[PAR] \quad \frac{P \xrightarrow{x} P'}{P|Q \xrightarrow{x} P'|Q}$$

$$[RES] \quad \frac{P \xrightarrow{x} P'}{(\text{new } y : r)\ P \xrightarrow{x} (\text{new } y : r)\ P'} \quad \text{if } x \neq y$$

$$[STOCH] \quad \frac{P \xrightarrow{x} P'}{(\text{new } x : r)\ P \xrightarrow{r} (\text{new } x : r)\ P'}$$

$$[REACT] \quad ((x(\vec{y}).P + M)|(\overline{x}\langle\vec{z}\rangle.Q + N)) \xrightarrow{x} (\{\vec{z}/\vec{y}\}P|Q)$$

Figure 1: Semantics of the StoPi-calculus.

Definition 3.1 (Process Normal Form)
For a process in the StoPi-calculus we define the normal form as

$$\text{new } (y_1 : r_1, \ldots, y_n : r_n)(P_1|\cdots|P_j) \qquad (3)$$

where each P_i is either $\sum_{l=0}^{n} \pi_k.P_k$ or $A(\vec{x})$.

The StoPi-calculus and Simulator
- a stochastic π-calculus and the implementation of a simulator

Anders Bloch bloch@cs.auc.dk
Bjørn Haagensen bh@cs.auc.dk
Michael Korsbæk Høyer fallen@cs.auc.dk
Steffen Ulsø Knudsen steffen@cs.auc.dk

Aalborg University, Department of Computer Science
Fredrik Bajers Vej 7E, DK-9220 Aalborg Øst

30th May 2003 (unpublished)
http://www.cs.auc.dk/~bh/files/dat4project.ps.gz

In the following let $c = S$ be a configuration, Ch_c the channels contained in the environments in the configuration c and assume that s and s' are summations in S. Furthermore let

$$|s|_{ch} = \left|\{(x, \vec{y}, P) \in A \mid s = (A, e) \wedge e(x) = ch\}\right|$$

and

$$|s|_{\overline{ch}} = \left|\{(\overline{x}, \vec{y}, P) \in A \mid s = (A, e) \wedge e(x) = ch\}\right|.$$

Then the apparent rate of a reaction on a channel in a configuration is defined as

Definition 3.4 (Apparent Rates)
The apparent rate of a reaction between guarded sums s and s' in a configuration c on channel $ch \in Ch_c$, where s receives input on ch and s' outputs on ch, is defined as:

$$r(s, s', ch) = \begin{cases} ch_r \cdot |s|_{ch} \cdot |s'|_{\overline{ch}} & \text{if } s \neq s' \\ 0 & \text{otherwise.} \end{cases} \qquad (6)$$

The apparent rate of channel ch in a configuration c is the sum of the apparent rates of a reactions on ch on all possible guarded sums:

$$r_{ch}(c) = \sum_{s,s' \in S} r(s, s', ch). \qquad (7)$$

Note that the expression (7) can explicitly be calculated as

$$r_{ch}(c) = ch_r \left(\sum_{s \in S} |s|_{ch} \cdot \sum_{\substack{s' \in S \\ s \neq s'}} |s'|_{\overline{ch}} \right). \qquad (8)$$

This is the most detailed analysis of a stochastic pi calculus (with mixed choice) yet. However, the apparent rate is defined on the underlying abstract machine after translation, not on the pi calculus itself.

SPiM

Reduction Semantics (with structural congruence)

$$P,Q ::= \nu x\, P \quad \text{Restriction} \quad (1)$$
$$\mid P \mid Q \quad \text{Parallel} \quad (2)$$
$$\mid \Sigma \quad \text{Summation} \quad (3)$$
$$\mid !\pi.P \quad \text{Replication} \quad (4)$$
$$\Sigma ::= \mathbf{0} \quad \text{Null} \quad (5)$$
$$\mid \pi.P + \Sigma \quad \text{Action} \quad (6)$$
$$\pi ::= x\langle n\rangle \quad \text{Output} \quad (7)$$
$$\mid x(m) \quad \text{Input}, x \neq m \quad (8)$$

Definition 1. *Syntax of* SPi

$$Q \equiv P \xrightarrow{r} P' \equiv Q' \Rightarrow Q \xrightarrow{r} Q' \quad (9)$$
$$P \xrightarrow{r} P' \Rightarrow \nu x\, P \xrightarrow{r} \nu x\, P' \quad (10)$$
$$P \xrightarrow{r} P' \Rightarrow P \mid Q \xrightarrow{r} P' \mid Q \quad (11)$$
$$x\langle n\rangle.P + \Sigma \mid x(m).Q + \Sigma' \xrightarrow{rate(x)} P \mid Q_{\{n/m\}} \quad (12)$$

Definition 2. *Reduction in* SPi

$$P \mid \mathbf{0} \equiv P \quad (13)$$
$$P \mid Q \equiv Q \mid P \quad (14)$$
$$P \mid (Q \mid R) \equiv (P \mid Q) \mid R \quad (15)$$
$$!\pi.P \equiv \pi.(P \mid !\pi.P) \quad (16)$$
$$\nu x\, \mathbf{0} \equiv \mathbf{0} \quad (17)$$
$$\nu x \nu y\, P \equiv \nu y \nu x\, P \quad (18)$$
$$\pi.P + \pi'.P' + \Sigma \equiv \pi'.P' + \pi.P + \Sigma \quad (19)$$
$$\Sigma \equiv \Sigma' \Rightarrow \pi.P + \Sigma \equiv \pi.P + \Sigma' \quad (20)$$
$$x \notin fn(P) \Rightarrow \nu x\,(P \mid Q) \equiv P \mid \nu x\, Q \quad (21)$$
$$P \equiv P' \Rightarrow \nu x\, P \equiv \nu x\, P' \quad (22)$$
$$P \equiv P' \Rightarrow P \mid Q \equiv P' \mid Q \quad (23)$$
$$P \equiv P' \Rightarrow !\pi.P \equiv !\pi.P' \quad (24)$$
$$P \equiv P' \Rightarrow \pi.P + \Sigma \equiv \pi.P' + \Sigma \quad (25)$$

Definition 3. *Structural congruence in* SPi

$$V,U ::= \nu x\, V \quad \text{Restriction} \quad (26)$$
$$\mid A \quad \text{List} \quad (27)$$
$$A,B ::= [] \quad \text{Empty} \quad (28)$$
$$\mid \Sigma :: A \quad \text{Summation} \quad (29)$$

Definition 7. *Syntax of* SPiM

$$n \notin fn(P) \Rightarrow P : (\nu x\, V) \triangleq \nu x\,(P : V) \quad (30)$$
$$\mathbf{0} : A \triangleq A \quad (31)$$
$$(P \mid Q) : A \triangleq P : Q : A \quad (32)$$
$$x \notin fn(P, A) \Rightarrow (\nu y\, P) : A \triangleq \nu x\,(P_{\{x/y\}} : A) \quad (33)$$
$$!\pi.P : A \triangleq \pi.(P \mid !\pi.P) :: A \quad (34)$$
$$(\pi.P + \Sigma) : A \triangleq (\pi.P + \Sigma) :: A \quad (35)$$

Definition 8. *Construction in* SPiM

$$(\!|P|\!) \triangleq P : [] \quad (36)$$

Definition 9. *Encoding* SPi *to* SPiM

Again, the apparent rate is defined on the underlying abstract machine.

$$V \xrightarrow{r} V' \Rightarrow \nu x\, V \xrightarrow{r} \nu x\, V' \quad (37)$$
$$\left.\begin{array}{l} x = Next(A) \\ \wedge A \succ (x(m).P + \Sigma) :: A' \\ \wedge A' \succ (x\langle n\rangle.Q + \Sigma') :: A'' \end{array}\right. \Rightarrow A \xrightarrow{rate(x)} P_{\{n/m\}} : Q : A'' \quad (38)$$

Definition 10. *Reduction in* SPiM

$$A @ \Sigma :: A' \succ \Sigma :: A @ A' \quad (39)$$
$$\Sigma :: A \succ (\pi'.P' + \Sigma') :: A \Rightarrow (\pi.P + \Sigma) :: A \succ (\pi'.P' + \pi.P + \Sigma') :: A \quad (40)$$

Definition 11. *Selection in* SPiM

Bio-CONCUR 2004 Preliminary Version

A Correct Abstract Machine for the Stochastic Pi-calculus

Andrew Phillips [1] Luca Cardelli [2]

The next reaction channel x and the reaction delay τ are calculated using the algorithm described in Definition 12. The algorithm is based on the Gillespie algorithm [6], which uses a notion of *channel activity* in order to stochastically select the next reaction channel. A similar notion of channel activity is defined for the SPi-Machine, where $\mathrm{Act}_x(A)$ denotes the activity of channel x in list A. The activity corresponds to the number of possible combinations of inputs and outputs on channel x in A, and is defined by:

$$\mathrm{Act}_x(A) = (\mathrm{In}_x(A) * \mathrm{Out}_x(A)) - \mathrm{Mix}_x(A)$$

where $\mathrm{In}_x(A)$ and $\mathrm{Out}_x(A)$ are the number of unguarded *inputs* and *outputs* on channel x in A, respectively, and $\mathrm{Mix}_x(A)$ = the sum of $\mathrm{In}_x(\Sigma_i) \times \mathrm{Out}_x(\Sigma_i)$ for each summation Σ_i in A. The formula takes into account the fact that an input and an output in the same summation cannot interact, by subtracting $\mathrm{Mix}_x(A)$ from the product of the number of inputs and outputs on x. Once the values x and τ have been calculated, the machine increments the reaction time by delay τ and randomly chooses one of the available reactions on x with equal probability, using the selection operator. This is achieved by randomly choosing a number $n \in [1..\mathrm{In}_x(A)]$ and selecting the nth input in A, followed by randomly selecting an output from the remaining list in a similar fashion.

Sπ – A modern version

Labeled Transition Semantics

$P, Q ::=$	$\nu x\, P$	Restriction	$E ::= X(m) = P$		Definition, $\mathrm{fn}(P) \subseteq m$
$\mid$	$P \mid Q$	Parallel	$\mid$	E_1, E_2	Union
$\mid$	M	Choice	$\mid$	$\emptyset$	Empty
$\mid$	$X(n)$	Instance			
			$\pi ::=$	$?x(m)$	Input
$M ::= \pi.P + M$		Action	$\mid$	$!x(n)$	Output
$\mid$	$\mathbf{0}$	Null	$\mid$	τ_r	Delay

A Graphical Representation for Biological Processes in the Stochastic pi-calculus

Andrew Phillips[1], Luca Cardelli[1], and Giuseppe Castagna[2]

$$R(x, P) = rate(x) \times (\mathrm{In}_x(P) \times \mathrm{Out}_x(P) - \mathrm{Mix}_x(P))$$

$$!x(n).P + M \xrightarrow{!x(n)} P \quad (1)$$

$$?x(m).P + M \xrightarrow{?x(n)} P_{\{n/m\}} \quad (2)$$

$$\tau_r.P + M \xrightarrow{r} P \quad (3)$$

$$P \xrightarrow{!x(n)} P' \quad Q \xrightarrow{?x(n)} Q' \quad \Rightarrow \quad P \mid Q \xrightarrow{x} P' \mid Q' \quad (4)$$

$$n \notin \mathrm{fn}(Q) \quad P \xrightarrow{!x(\nu n)} P' \quad Q \xrightarrow{?x(n)} Q' \quad \Rightarrow \quad P \mid Q \xrightarrow{x} \nu n\,(P' \mid Q') \quad (5)$$

$$P \xrightarrow{x} P' \quad \Rightarrow \quad \nu x\, P \xrightarrow{R(x,P)} \nu x\, P' \quad (6)$$

$$x \neq y \quad P \xrightarrow{!x(y)} P' \quad \Rightarrow \quad \nu y\, P \xrightarrow{!x(\nu y)} P' \quad (7)$$

$$x \notin \mathrm{fn}(\alpha) \cup \mathrm{bn}(\alpha) \quad P \xrightarrow{\alpha} P' \quad \Rightarrow \quad \nu x\, P \xrightarrow{\alpha} \nu x\, P' \quad (8)$$

$$M \xrightarrow{\alpha} P' \quad \Rightarrow \quad \pi.P + M \xrightarrow{\alpha} P' \quad (9)$$

$$\mathrm{bn}(\alpha) \cap \mathrm{fn}(Q) = \emptyset \quad P \xrightarrow{\alpha} P' \quad \Rightarrow \quad P \mid Q \xrightarrow{\alpha} P' \mid Q \quad (10)$$

$$X(m) = P \quad P_{\{n/m\}} \xrightarrow{\alpha} P' \quad \Rightarrow \quad X(n) \xrightarrow{\alpha} P' \quad (11)$$

Sπ – My favorite version

Reduction Semantics
(with structural congruence)

[A.Phillips]

$$\tau_r.P + M \xrightarrow{r} P \tag{126}$$

$$!x(n).P + M \mid ?x(m).Q + N \xrightarrow{x} P \mid Q_{\{n/m\}} \tag{127}$$

$$P \xrightarrow{x} P' \Rightarrow \nu x\, P \xrightarrow{R(x,P)} \nu x\, P' \tag{128}$$

$$\theta \neq x \quad P \xrightarrow{\theta} P' \Rightarrow \nu x\, P \xrightarrow{\theta} \nu x\, P' \tag{129}$$

$$P \xrightarrow{\theta} P' \Rightarrow P \mid Q \xrightarrow{\theta} P' \mid Q \tag{130}$$

$$Q \equiv P \xrightarrow{\theta} P' \equiv Q' \Rightarrow Q \xrightarrow{\theta} Q' \tag{131}$$

Reduction in Sπ, where $\theta ::= x|r$ and $R(x,P)$ is the apparent rate of x in P.

$P,Q ::=$	$\nu x\, P$	Restriction	$E ::=$	$X(m)=P$	Definition, $\mathrm{fn}(P) \subseteq m$
$\mid$	$P \mid Q$	Parallel	$\mid$	E_1, E_2	Union
$\mid$	M	Choice	$\mid$	$\emptyset$	Empty
$\mid$	$X(n)$	Instance			
			$\pi ::=$	$?x(m)$	Input
$M ::=$	$\pi.P + M$	Action	$\mid$	$!x(n)$	Output
$\mid$	$\mathbf{0}$	Null	$\mid$	τ_r	Delay

Definition 1.1. Syntax of Sπ

$$P \mid \mathbf{0} \equiv P \tag{102}$$
$$P \mid Q \equiv Q \mid P \tag{103}$$
$$P \mid (Q \mid R) \equiv (P \mid Q) \mid R \tag{104}$$
$$\pi_1.P_1 + \pi_2.P_2 + M \equiv \pi_2.P_2 + \pi_1.P_1 + M \tag{105}$$
$$\nu x\, \mathbf{0} \equiv \mathbf{0} \tag{106}$$
$$\nu x\, \nu y\, P \equiv \nu y\, \nu x\, P \tag{107}$$
$$x \notin \mathrm{fn}(P) \Rightarrow \nu x\, (P \mid Q) \equiv P \mid \nu x\, Q \tag{108}$$
$$X(m) \triangleq P \Rightarrow X(n) \equiv P_{\{n/m\}} \tag{109}$$

Definition 1.2. Structural Congruence in Sπ

$$\mathrm{In}_x(\nu x\, P) \triangleq 0 \tag{110}$$
$$x \neq y \Rightarrow \mathrm{In}_x(\nu y\, P) \triangleq \mathrm{In}_x(P) \tag{111}$$
$$\mathrm{In}_x(P \mid Q) \triangleq \mathrm{In}_x(P) + \mathrm{In}_x(Q) \tag{112}$$
$$X(m)=P \Rightarrow \mathrm{In}_x(X(n)) \triangleq \mathrm{In}_x(P_{\{n/m\}}) \tag{113}$$
$$\mathrm{In}_x(\pi.P + M) \triangleq \mathrm{In}_x(\pi) + \mathrm{In}_x(M) \tag{114}$$
$$\mathrm{In}_x(\mathbf{0}) \triangleq 0 \tag{115}$$

$$\mathrm{In}_x(?x(m)) \triangleq 1 \tag{116}$$
$$x \neq y \Rightarrow \mathrm{In}_x(?y(m)) \triangleq 0 \tag{117}$$
$$\mathrm{In}_x(!y(n)) \triangleq 0 \tag{118}$$
$$\mathrm{In}_x(\tau_r) \triangleq 0 \tag{119}$$

$$\mathrm{Mix}_x(\nu x\, P) \triangleq 0 \tag{120}$$
$$x \neq y \Rightarrow \mathrm{Mix}_x(\nu y\, P) \triangleq \mathrm{Mix}_x(P) \tag{121}$$
$$\mathrm{Mix}_x(P \mid Q) \triangleq \mathrm{Mix}_x(P) + \mathrm{Mix}_x(Q) \tag{122}$$
$$X(m)=P \Rightarrow \mathrm{Mix}_x(X(n)) \triangleq \mathrm{Mix}_x(P_{\{n/m\}}) \tag{123}$$
$$\mathrm{Mix}_x(M) \triangleq \mathrm{In}_x(M) \times \mathrm{Out}_x(M) \tag{124}$$
$$R(x,P) = rate(x) \times (\mathrm{In}_x(P) \times \mathrm{Out}_x(P) - \mathrm{Mix}_x(P)) \tag{125}$$

Definition 1.3. Apparent Rate in Sπ. The definition of $\mathrm{Out}_x(P)$ is similar to that of $\mathrm{In}_x(P)$.

Proposition 1.4. $\forall P.P \in S\pi \wedge P \equiv Q \Rightarrow R(x,P) = R(x,Q)$

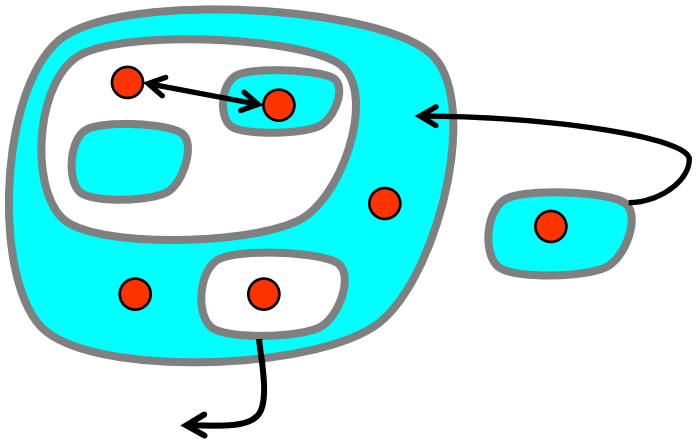
BioAmbients

- An extension of Sto-π-calculus
  - Dynamic membranes: operations for merging, splitting, interacting through membrane channels.
  - Implemented by Aviv Regev.

- An adaptation of Ambient Calculus
  - A process language for dynamic containers (mobile agents, distributed locations, etc.)

BioAmbients: An abstraction for biological compartments

Aviv Regev [a,*] Ekaterina M. Panina [b] William Silverman [c]
Luca Cardelli [d] Ehud Shapiro [c]



Processes can communicate across membranes

Membranes are processes; they can move in and out of other membranes

Sπ to CTMCs

- Theorem [Hillston, Priami]
  - Let P be a *finite control* Sπ process.
    Then the set of derivative processes from P defines a finite CTMC.

Gillespie

1. For all $x \in fn(A)$ calculate $a_x = \mathrm{Act}_x(A) * rate(x)$
2. Store non-zero values of a_x in a list (x_μ, a_μ), where $\mu \in 1...M$.
3. Calculate $a_0 = \sum_{\nu=0}^{M} a_\nu$
4. Generate two random numbers $n_1, n_2 \in [0,1]$ and calculate τ, μ such that:

$$\tau = (1/a_0)\ln(1/n_1)$$

$$\sum_{\nu=1}^{\mu-1} a_\nu < n_2 a_0 \leq \sum_{\nu=1}^{\mu} a_\nu$$

5. $Next(A) = x_\mu$ and $Delay(A) = \tau$.

Definition 12. *Calculating $Next(A)$ and $Delay(A)$ according to Gillespie [6].*

2340

Exact Stochastic Simulation of Coupled Chemical Reactions

Daniel T. Gillespie*

Research Department, Naval Weapons Center, China Lake, California 93555 *(Received May 12, 1977)*

Publication costs assisted by the Naval Weapons Center

The Journal of Physical Chemistry, Vol. 81, No. 25, 1977

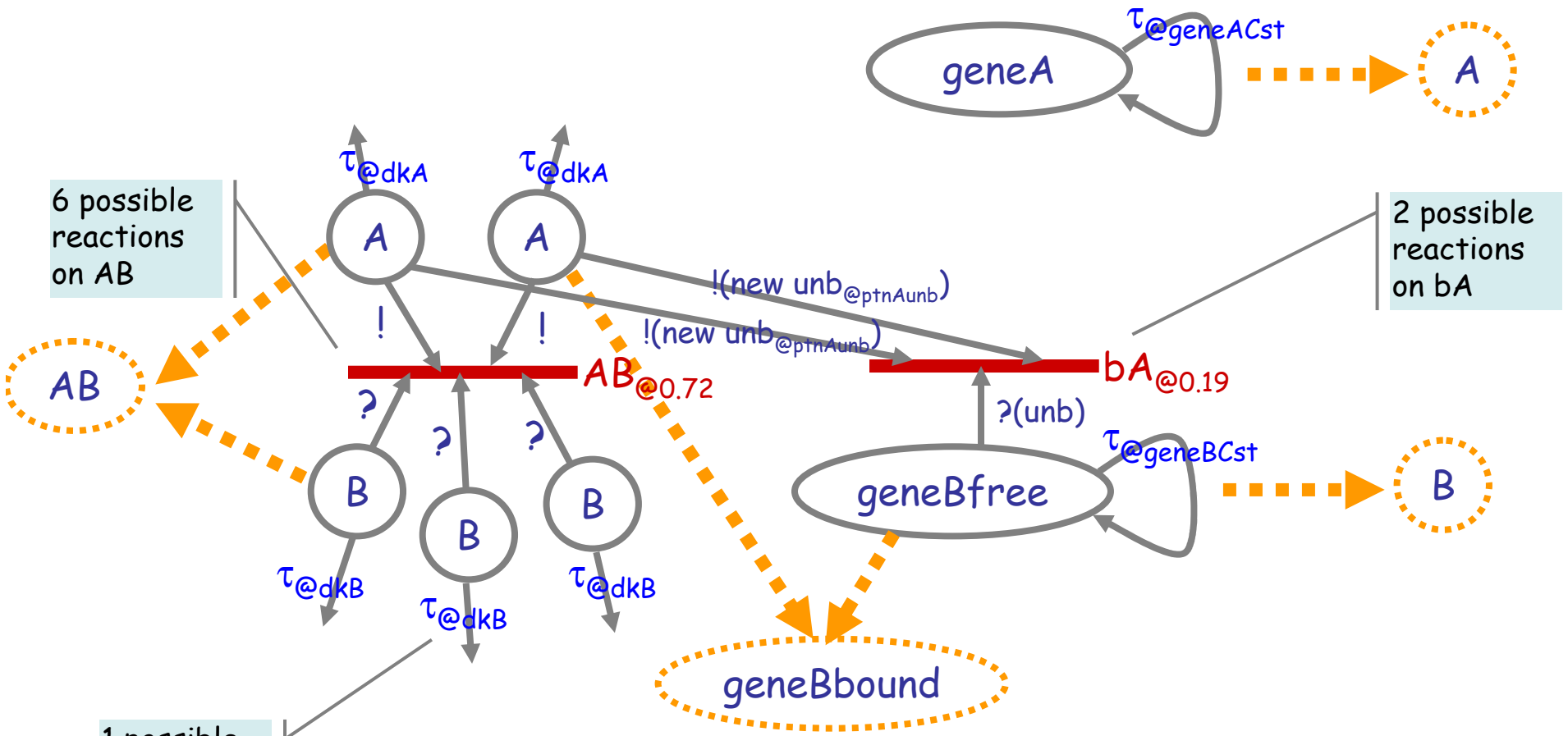
A Correct Abstract Machine for the Stochastic Pi-calculus

Andrew Phillips [1] Luca Cardelli [2]

that's Area 51!

One State of the Simulation

(François and Hakim example)



Use Gillespie to pick a reaction and move on

Summary

- Stochastic process calculi:
  - The modular representation of concurrent stochastic processes.
  - Cut down to CTMCs (Continuous Time Markov Chains). Then, standard analysis techniques are applicable.
  - Can be given friendly automata-like scalable graphical syntax [A.Phillips].
  - Is directly executable (via Gillespie).
  - Are analyzable [Hillston] [Hermanns].

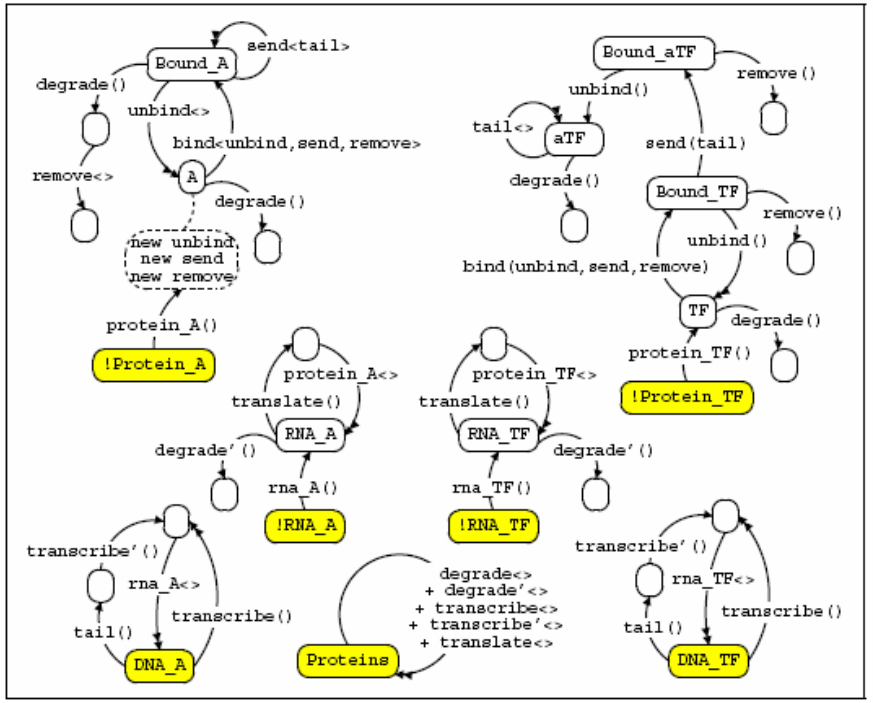


Figure 2. Regulating Gene Expression by Positive Feedback [9]

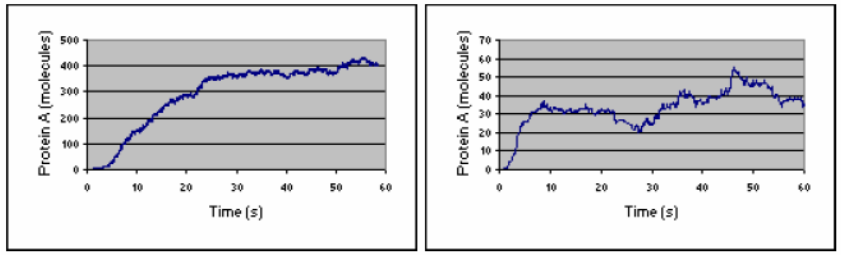
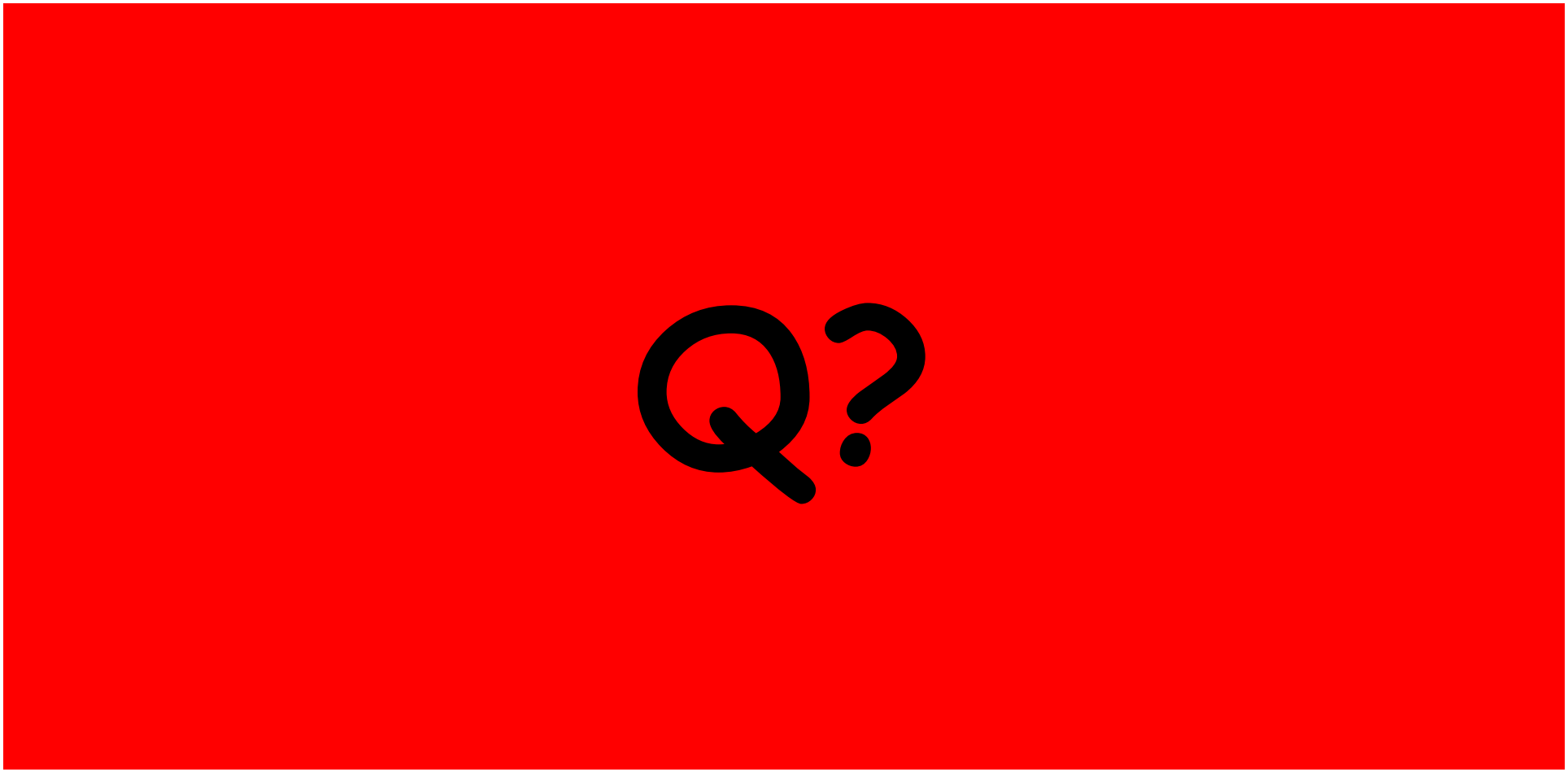


Figure 3. Protein A molecules v.s. time in presence (left) and absence (right) of TF

A.Phillips, L.Cardelli. BioConcur'04.


Q?