
A Probabilistic Calculus for Molecular Systems

Roberto Barbuti, Stefano Cataudella, Andrea Maggiolo-Schettini,
Paolo Milazzo, and Angelo Troina

Department of Computer Science, University of Pisa (Italy)

Introduction

- Formal models for systems of interactive components can be easily adapted for the modeling of biochemical phenomena
 - The modeling of biochemical reactions allows:
 1. the development of simulators
 2. the model checking of properties of biochemical systems
 3. (hopefully) the prediction of some unknown kinetic constants
 - In this work:
 1. we introduce a probabilistic calculus for molecular reactions
 2. we use the calculus for modeling an example of real enzymatic activity
 3. we report some experimental results of simulation and model checking
-

A calculus of solutions (1)

- Let \mathcal{E} be an infinite set of elementary particles:

(molecules) $m ::= X \mid m-m$

(solutions) $S ::= \mathbf{0} \mid m \mid S, S$

where X is any particle of \mathcal{E} and $\mathbf{0}$ is the empty solution.

- The *structural congruence* for solutions \equiv is the smallest equivalence relation such that:

$$S_1, S_2 \equiv S_2, S_1 \quad (S_1, S_2), S_3 \equiv S_1, (S_2, S_3) \quad S, \mathbf{0} \equiv S$$

$$m_1-m_2 \equiv m_2-m_1 \quad (m_1-m_2)-m_3 \equiv m_1-(m_2-m_3)$$

A calculus of solutions (2)

$$m ::= X \mid m-m$$

$$S ::= \mathbf{0} \mid m \mid S, S$$

$$S_1, S_2 \equiv S_2, S_1 \quad S, 0 \equiv S$$

$$(S_1, S_2), S_3 \equiv S_1, (S_2, S_3)$$

$$m_1-m_2 \equiv m_2-m_1$$

$$(m_1-m_2)-m_3 \equiv m_1-(m_2-m_3)$$

- A solution can be considered as a multi-set of molecules
 - For instance, \in can be defined as follows:

$$m \in S \iff \exists S' \text{ s.t. } m, S' \equiv S$$

Probabilistic rules

- A *probabilistic rule* (or *reaction*) is a triple

$$(S, P, S')$$

where S and S' are solutions and P is a total function from the infinite set of molecules into $[0, 1[$

- Examples (assume $P(m) = 0$ if not specified):

$$R_1 = (\{a, b\}, \{a \mapsto \frac{2}{3}, b \mapsto \frac{2}{3}\}, \{a-b\})$$

$$R_2 = (\{a-b\}, \{a-b \mapsto \frac{1}{2}\}, \{a, b\})$$

- A *system* is a pair (S, \mathcal{R}) where S is a solution and \mathcal{R} is a finite set of probabilistic rules
 - Example: $(\{a, b\}, \{R_1, R_2\})$
-

Well-formed set of rules

- A set of rules $\mathcal{R} = \{(S_1, P_1, S'_1), \dots, (S_n, P_n, S'_n)\}$ is well-formed if it satisfies the following:
 - for all $(S_i, P_i, S'_i) \in \mathcal{R}$
 - $S_i \cap S'_i = \emptyset$
 - $\#(S_i) = \#(S'_i)$
 - $P_i(m) > 0$ if $m \in S_i$, $P_i(m) = 0$ otherwise
 - if $m \equiv m'$ then $P_i(m) = P_i(m')$
 - for any molecule m
 - $0 < P^I(m) \stackrel{\text{def}}{=} (1 - \sum_{i=0}^n P_i(m))$
 - for all $i, j \in 1, \dots, n$ if $i \neq j$ then either $S_i \neq S_j$ or $S'_i \neq S'_j$

Example of well-formed set of rules

$$\mathcal{R} = \{R_1, R_2\}$$
$$R_1 = (\{a, b\}, \{a \mapsto \frac{2}{3}, b \mapsto \frac{2}{3}\}, \{a-b\})$$
$$R_2 = (\{a-b\}, \{a-b \mapsto \frac{1}{2}\}, \{a, b\})$$

- for all $(S_i, P_i, S'_i) \in \mathcal{R}$
 - $S_i \cap S'_i = \emptyset$
 - $\#(S_i) = \#(S'_i)$
 - $P_i(m) > 0$ if $m \in S_i$, $P_i(m) = 0$ otherwise
 - if $m \equiv m'$ then $P_i(m) = P_i(m')$
- for any molecule m
 - $0 < P^I(m) \stackrel{\text{def}}{=} (1 - \sum_{i=0}^n P_i(m))$
- for all $i, j \in 1, \dots, n$ if $i \neq j$ then either $S_i \neq S_j$ or $S'_i \neq S'_j$

Step semantics

- Given \mathcal{R} , we denote with \mathcal{R}^I the (not well-formed) set of rules:

$$\mathcal{R}^I = \{(\{m\}, \{m \mapsto P^I(m)\}, \{m\}) \mid m \text{ is a molecule}\}$$

- The following recursive algorithm $Step(S)$ describes how a system (S, \mathcal{R}) executes a step (it also returns the solution for the next step):

1. **choose** m in S

} RANDOMLY CHOOSE
A MOLECULE m IN S

2. **let** R_1, \dots, R_n be the only rules in $\mathcal{R} \cup \mathcal{R}^I$
such that $R_i = (S_i, P_i, S'_i)$, $m \in S_i$, $S_i \subseteq S$

} CONSIDER ONLY THE
RULES (IN $\mathcal{R} \cup \mathcal{R}^I$) THAT CAN BE
APPLIED TO m IN S

3. **choose** $R = (S_R, P_R, S'_R)$ in R_1, \dots, R_n
with probabilities $\frac{P_1(m)}{\sum_{i=1}^n P_i(m)}, \dots, \frac{P_n(m)}{\sum_{i=1}^n P_i(m)}$

} RANDOMLY CHOOSE ONE OF THESE
RULES AFTER NORMALIZING
THEIR PROBABILITIES

4. **if** $S \setminus S_R = \emptyset$ **then return** S'_R
else return $(S'_R \cup Step(S \setminus S_R))$

} RECURSIVELY EXECUTE ON
THE REST OF THE SOLUTION

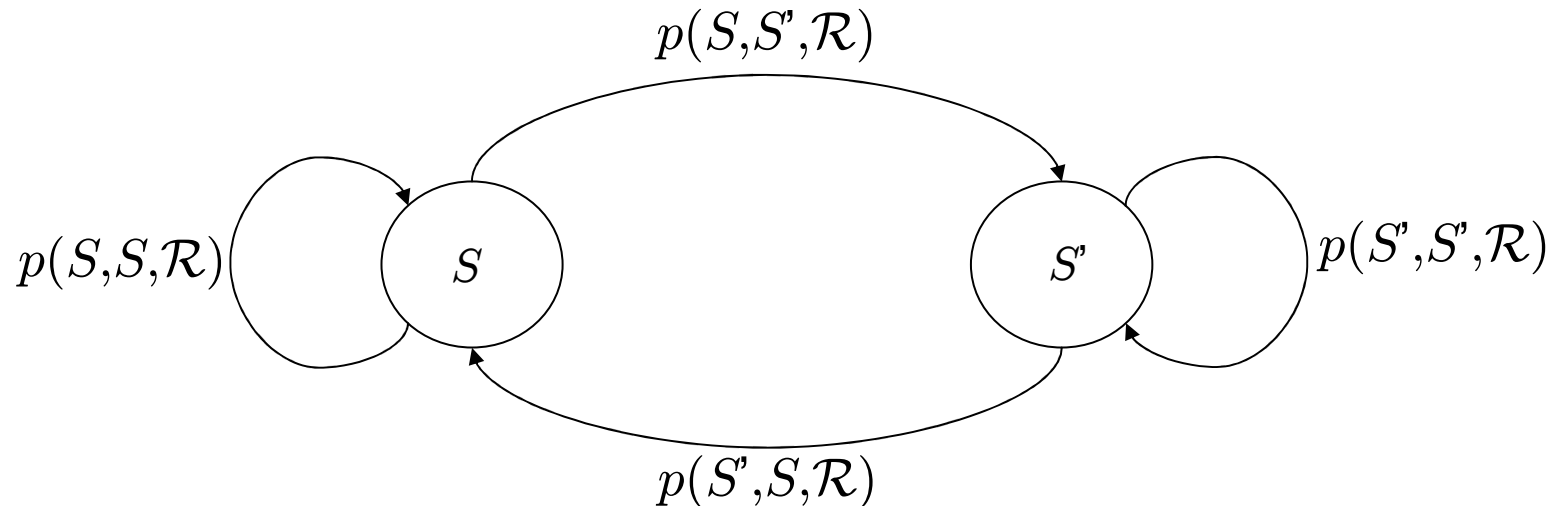
Molecular Probabilistic Transition

System (1)

- Given two solutions S, S' and a set of rules \mathcal{R} , it is possible to infer from the algorithm the probability of S to be transformed into S'

$$p(S, S', \mathcal{R})$$

- Moreover, it is possible to build a probabilistic transition system:



Molecular Probabilistic Transition System (2)

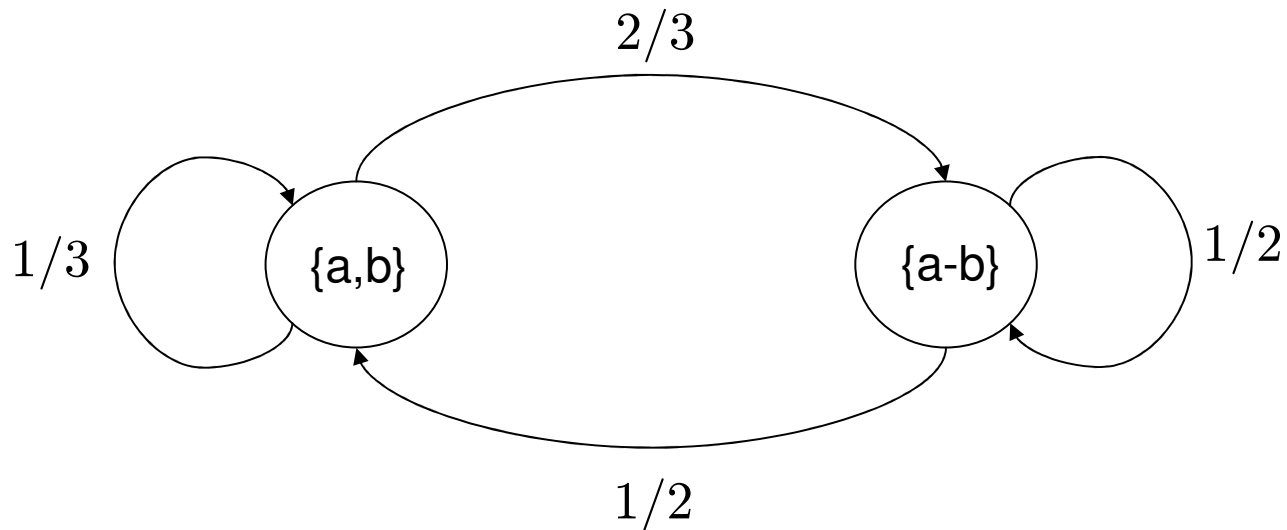
- Given a system (S, \mathcal{R}) the *Molecular Probabilistic Transition System (MPTS)* of the system is a tuple $M = (Q, S, \mathcal{R}, \delta, \pi)$ where:
 - Q is a finite set of solutions of density equal to $\#(S)$
 - S is the initial solution (the solution of the system)
 - \mathcal{R} is the set of rules of the system
 - $\delta \subseteq Q \times Q$ is a finite set of transitions
 - $\pi : \delta \rightarrow [0, 1]$ is a probability function such that if $e = (S_1, S_2) \in \delta$, then $\pi(e) = p(S_1, S_2, \mathcal{R})$
 - (Correctness) Given a system (S, \mathcal{R}) and its MPTS $M = (Q, S, \mathcal{R}, \delta, \pi)$, there is $e = (S_1, S_2) \in \delta$ with $\pi(e) = p(S_1, S_2, \mathcal{R})$ if and only if there is $S_2 = \text{Step}(S_1)$
-

Example of MPTS

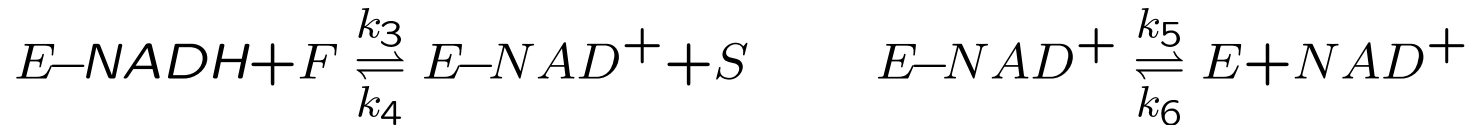
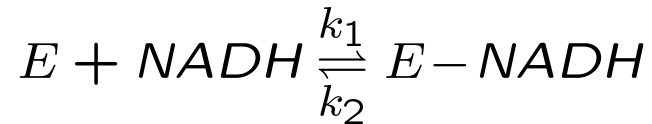
System: $(S = \{a, b\}, \mathcal{R} = \{R_1, R_2\})$

where $R_1 = (\{a, b\}, \{a \mapsto \frac{2}{3}, b \mapsto \frac{2}{3}\}, \{a-b\})$

$R_2 = (\{a-b\}, \{a-b \mapsto \frac{1}{2}\}, \{a, b\})$



Application: Sorbitol dehydrogenase



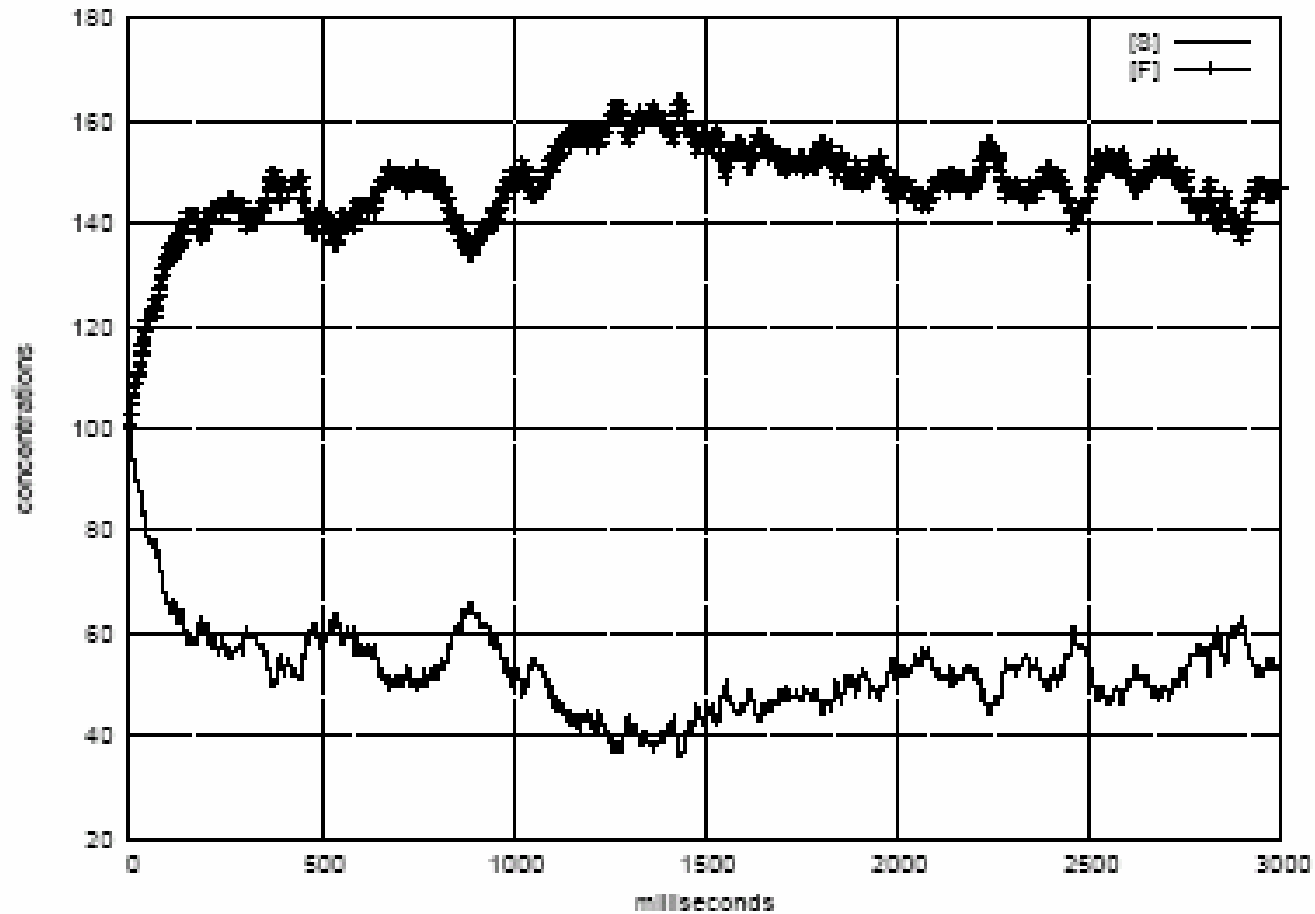
- \mathcal{R} {
1. ({E,NADH} , {(E,0.468),(NADH,0.862)} , {E-NADH})
 2. ({E-NADH,F} , {(E-NADH,0.001),(F,0.002)} , {E-NAD⁺,S})
 3. ({E-NAD⁺} , {(E-NAD⁺,0.774)} , {E,NAD⁺})
 4. ({E,NAD⁺} , {(E,0.457),(NAD⁺,0.859)} , {E-NAD⁺})
 5. ({E-NAD⁺,S} , {(E-NAD⁺,0.002),(S,0.008)} , {E-NADH,F})
 6. ({E-NADH} , {(E-NADH,0.772)} , {E,NADH})
- \mathcal{R}^I {
7. ({E} , {(E,0.075)} , {E})
 8. ({S} , {(S,0.992)} , {S})
 9. ({F} , {(F,0.998)} , {F})
 10. ({NADH} , {(NADH,0.138)} , {NADH})
 11. ({NAD⁺} , {(NAD⁺,0.141)} , {NAD⁺})
 12. ({E-NADH} , {(E-NADH,0.227)} , {E-NADH})
 13. ({E-NAD⁺} , {(E-NAD⁺,0.254)} , {E-NAD⁺})

Simulation (1)

- Simulator implemented in SICStus Prolog
- Given a system (S, R) the simulator
 - executes $S' = Step(S)$
 - gives the concentrations of the molecules in S' as output
 - iterates on (S', \mathcal{R})
- Sorbitol dehydrogenase: initial concentrations

E	S	F	NADH	NAD ⁺	E-NADH	E-NAD ⁺
10	100	100	100	100	0	0

Simulation (2)



Model checking (1)

- A probabilistic model checker can be easily used for verifying properties of a Molecular Probabilistic Transition System
- We used the PRISM model checker for verifying properties of the sorbitol dehydrogenase
- Properties have been specified in the CSL language:

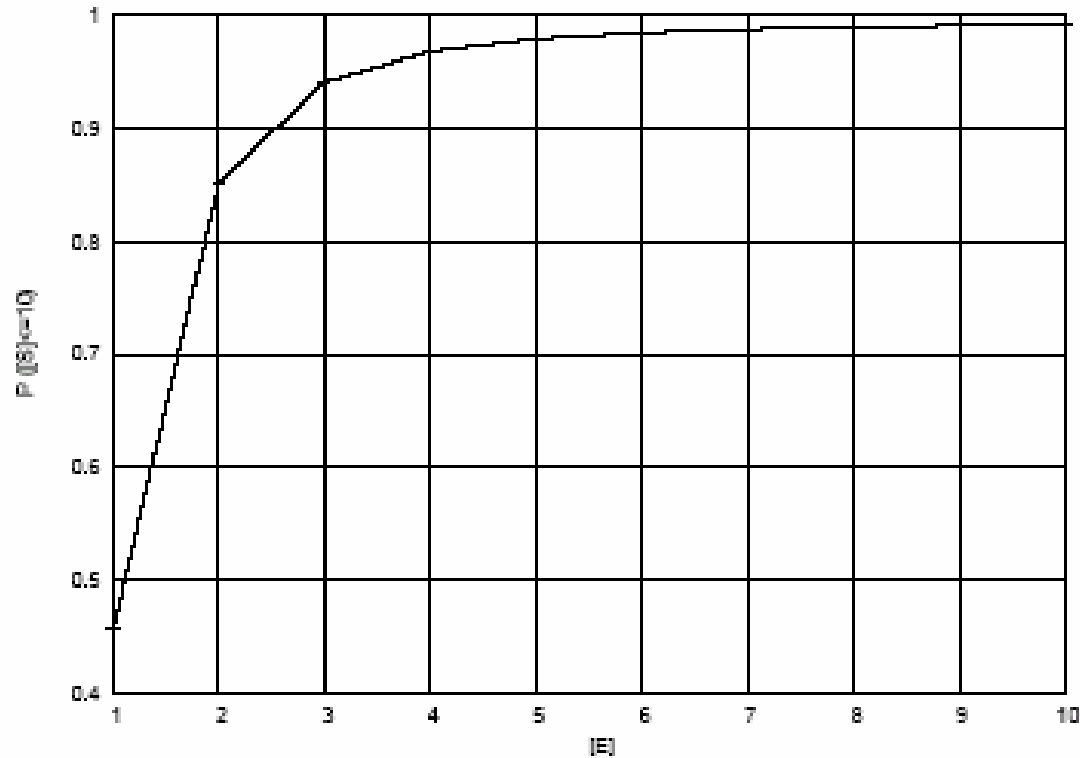
$$\phi ::= true \mid false \mid a \mid \phi \wedge \phi \mid \phi \vee \phi \mid \neg \phi \mid \mathcal{P}_{\sim p}[\psi] \mid \mathcal{S}_{\sim p}[\psi]$$

$$\psi ::= X\phi \mid \phi \mathcal{U}^I \phi \mid \phi \mathcal{U} \phi$$

- Sorbitol dehydrogenase: initial concentrations

S	F	NADH	NAD ⁺	E-NADH	E-NAD ⁺
25	25	25	25	0	0

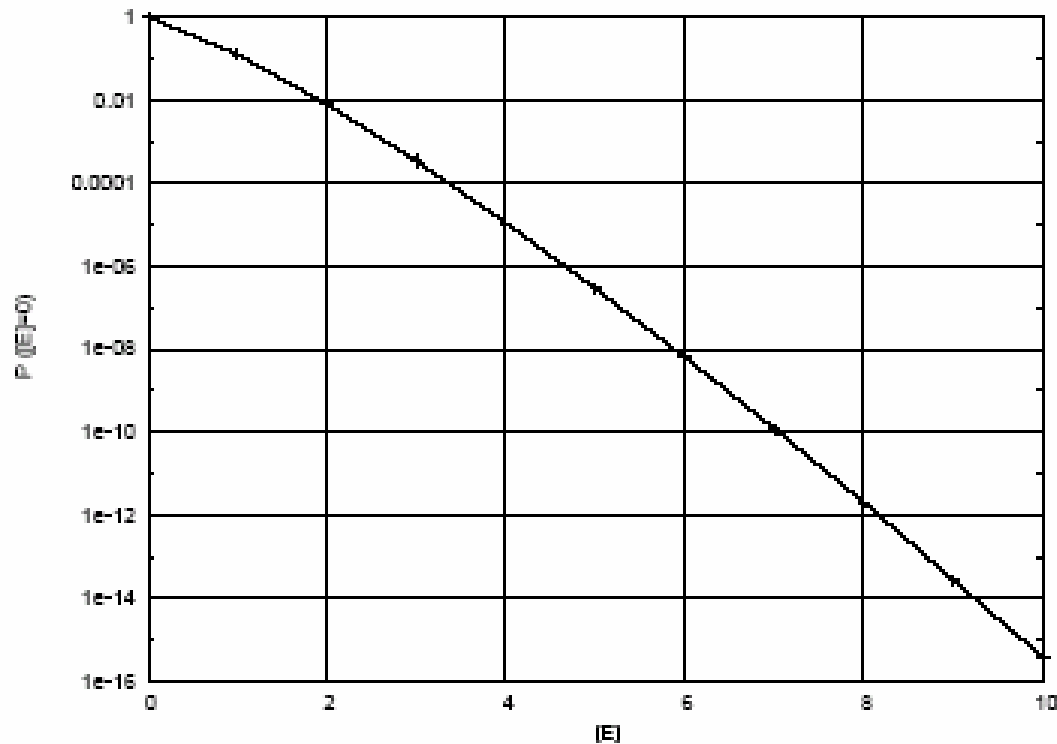
Model checking (2)



- Probability that the concentration of S becomes less than 10 in less than 5000 steps by varying the initial concentration of E

$$\mathcal{P}=? [\text{true} \mathcal{U}^{\leq 5000} [S] \leq 10]$$

Model checking (3)



- Probability that the concentration of E reaches 0 in less than 5000 steps by varying its initial value

$$\mathcal{P}_{=?} [\text{true} \mathcal{U}^{\leq 5000} [E] = 0]$$

Conclusions

- We have developed a framework to model biochemical reactions
 - we have implemented a simulator
 - we have used a model checker (PRISM) to verify properties of reactions
 - The main idea is to approximate the speed of reactions by probabilities of rule applications
 - The choice of the right probabilities is critical !!
 - Experiments have suggested to revise the semantics of the calculus to have a better approximation
-

Future work

- Expected improvements (short term):
 - the model will be revised in order to obtain better quantitative results
 - the choice of the probabilities will not be done by the modeler but probabilities will be derived automatically by kinetic constants
 - the semantics will be given in SOS style
 - some results of biological correctness will be proved

 - Expected improvements (mid/long term):
 - more complex examples of real biological systems will be simulated and verified
 - some additional operators will be considered
 - the problem of predicting unknown kinetic constants will be faced
-

THE END...

Computing probabilities...

$$p(m) = \frac{[m]_S}{|S|} \qquad p(R_{i,1 \leq i \leq n} | m) = \frac{P_i(m)}{P_1(m) + \dots + P_n(m)}$$

$$p(S, S', \mathcal{R}) = \begin{cases} 1 & \text{if } S = \emptyset = S' \\ 0 & \text{if } \#(S) \neq \#(S') \\ \sum_{i=1}^k p(m_i) \sum_{j=1}^{n_i} p(R_{i,1 \leq i \leq n_i} | m_i) p(S \setminus S_j, S' \setminus S'_j, \mathcal{R}) & \text{otherwise} \end{cases}$$

Model checking (1)

- A probabilistic model checker can be easily used for verifying properties of a Molecular Probabilistic Transition System
- We used the PRISM model checker for verifying properties of the sorbitol dehydrogenase
- Properties have been specified in the CSL language:

$$\phi ::= true \mid false \mid a \mid \phi \wedge \phi \mid \phi \vee \phi \mid \neg \phi \mid \mathcal{P}_{\sim p}[\psi] \mid \mathcal{S}_{\sim p}[\psi]$$

$$\psi ::= X\phi \mid \phi \mathcal{U}^I \phi \mid \phi \mathcal{U} \phi$$

- Sorbitol dehydrogenase: initial concentrations

S	F	NADH	NAD ⁺	E-NADH	E-NAD ⁺
25	25	25	25	0	0