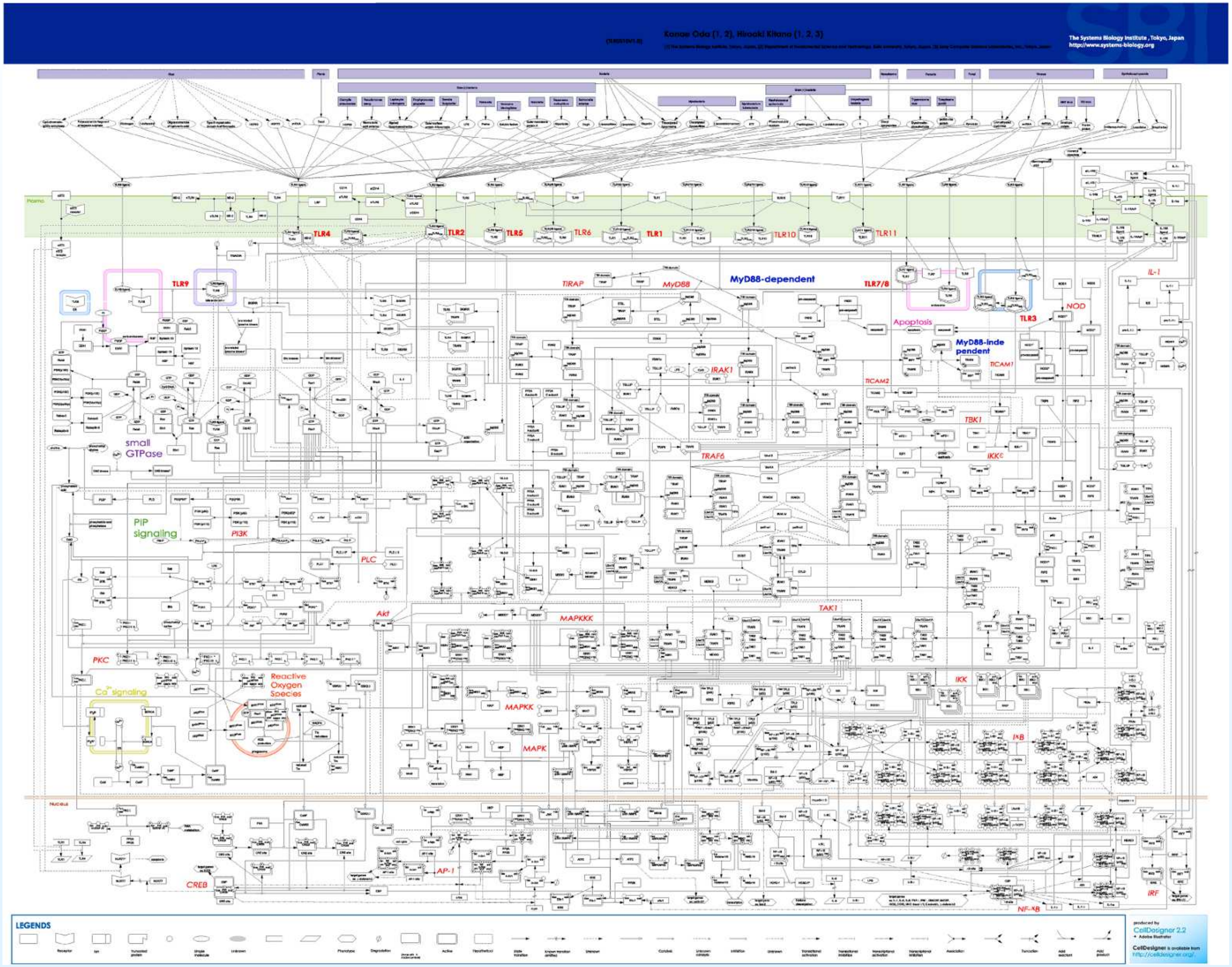
Efficient Analysis of Dynamical Properties in Stochastic Chemical Kinetic Models

Hiroyuki Kuwahara
Lane Center for Computational Biology
Carnegie Mellon University

CMACS

April 2, 2010

A Detailed Schematic Diagram of a Biological System



Model

- An abstraction of reality.
- Cannot capture everything.
- Useful models:
  - Explain things.
  - Predict things.
- Sufficient details are needed.
- Do we want to model an ecological system at the molecular level?
- Needs to balance accuracy and efficiency.
- Make things as simple as possible but not simpler.

Detailed View

C. Jordan, Gyre, 2009

Higher Level View

C. Jordan, Gyre, 2009

Global View

C. Jordan, Gyre, 2009

Stochastic Formations of Biochemical Models

- *Molecular Dynamics*:
  - Keeps track of positions and velocities of all the molecules.
  - Captures both reactive and non-reactive collisions as well as movements of diffusing molecules.
- *Green's Function Reaction Dynamics*:
  - Keeps track of a set of diffusing molecules.
  - Captures both reactive and non-reactive collisions of molecules via discrete events.
- *Stochastic Chemical Kinetics*:
  - Keeps track of molecular populations.
  - Captures only reactive collisions via discrete events.

Stochastic Chemical Kinetics (SCK)

Considers molecules of N species $\{S_1, \ldots, S_N\}$, interacting through M reaction channels $\{R_1, \ldots, R_M\}$ inside a well-stirred system.

- $\mathbf{X}(t) = (X_1(t), \ldots, X_N(t))$ is the system state that denotes the number of molecules of each S_i in the system at time t.
- Given $\mathbf{X}(t) = \mathbf{x}$, each reaction R_j is characterized by:
  - Propensity function $a_j(\mathbf{x})$ where $a_j(\mathbf{x})dt$ is probability that one R_j event will occur within next dt.
  - State-change vector $\mathbf{v_j}$ where one R_j event results in state transition $\mathbf{x} \rightarrow \mathbf{x} + \mathbf{v_j}$.

Time Evolution of SCK Models

Given $\mathbf{X}(t_0) = \mathbf{x_0}$, the time evolution of SCK model is governed by:

$$\mathbf{X}(t + dt) = \mathbf{X}(t) + \Xi(dt; \mathbf{X}(t)),$$

where $\Xi(dt; \mathbf{x})$ is a random variable with density function $p_\Xi(\mathbf{v} \mid dt; \mathbf{x})$:

$$p_\Xi(\mathbf{v} \mid dt; \mathbf{x}) = \begin{cases} a_j(\mathbf{x})dt & \text{if } \mathbf{v} = \mathbf{v}_j, \\ 1 - \sum_{j'=1}^{M} a_{j'}(\mathbf{x})dt & \text{if } \mathbf{v} = \mathbf{0}. \end{cases}$$

- Ignores the case where two or more reactions occur in time interval $[t, t + dt)$ as this probability is proportional to $(dt)^2$ (i.e., very small).
- Strictly speaking, each reaction must be elementary.

Simulation of SCK Models (Naive Approach)

Replace dt by small but finite value Δt:

$$\mathbf{X}(t + \Delta t) = \mathbf{X}(t) + \Xi(\Delta t; \mathbf{X}(t)).$$

- Not exact since Δt is finite.
- Not efficient since Δt must be very small.

Gillespie's Stochastic Simulation Algorithm (SSA)

Idea: Don't approximate dt by Δt, but instead, randomly sample the waiting time to the next reaction $T(\mathbf{x})$ and the next reaction index $J(\mathbf{x})$.

It turns out:

- $T(\mathbf{x})$ is an exponential random variable with mean $1/\sum_{j'} a_{j'}(\mathbf{x})$.
- $J(\mathbf{x})$ is a random variable with $Prob(j \mid \mathbf{x}) = a_j(\mathbf{x})/\sum_{j'} a_{j'}(\mathbf{x})$.

```
initialize: $t \leftarrow 0$, $\mathbf{x} \leftarrow \mathbf{x_0}$
evaluate all propensity functions.
**repeat**
    generate $\tau$ and $j$ according to $P(j, \tau \mid \mathbf{x}, t)$
    update: $t \leftarrow t + \tau$, $\mathbf{x} \leftarrow \mathbf{x} + \mathbf{v_j}$
    evaluate propensity functions of reactions affected by the change.
**until** simulation termination condition is satisfied
```

Simple Example: Enzymatic Reaction

$$
\begin{aligned}
R_1 &: \; E + S \xrightarrow{k_1} C, & a_1(\mathbf{x}) &= k_1 x_S x_E \\
R_2 &: \; C \xrightarrow{k_2} E + S, & a_2(\mathbf{x}) &= k_2 x_C \\
R_3 &: \; C \xrightarrow{k_3} E + P, & a_3(\mathbf{x}) &= k_3 x_C
\end{aligned}
$$

- Three reaction channels.
- Transforms S into P, catalyzed by E.

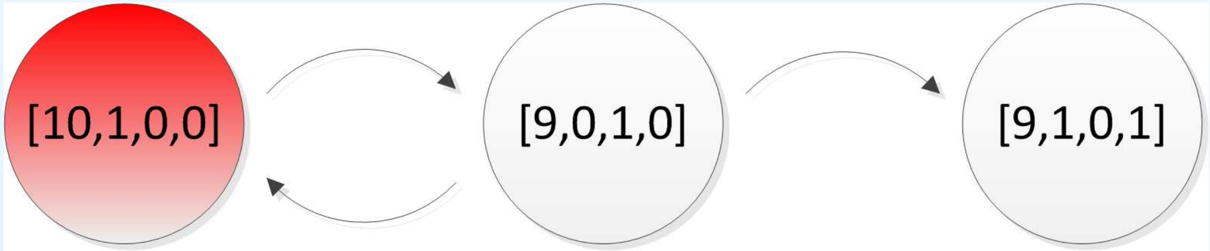
Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 10$	10
R_2	$k_2 x_C = 0$	10
R_3	$k_3 x_C = 0$	10

$$r_1 = 0.00475, \quad r_2 = 0.420$$
$$\tau = -\ln(r_1)/(10+0+0) = 0.535$$
$$\theta = r_2 \times (10+0+0) = 4.200$$

Iteration 1



$$t = 0$$

Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

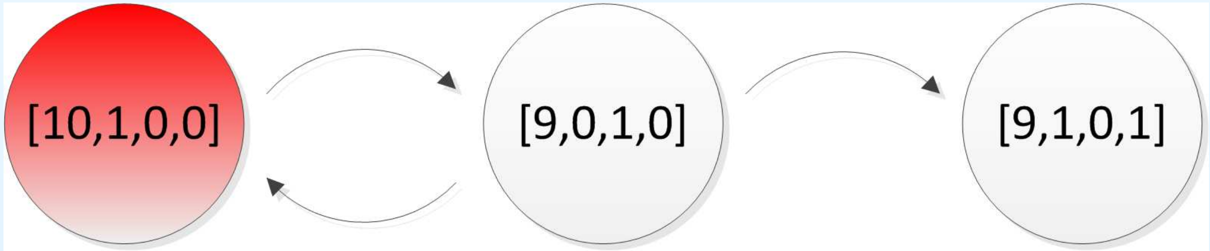
Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 10$	10
R_2	$k_2 x_C = 0$	10
R_3	$k_3 x_C = 0$	10

$$r_1 = 0.00475, \quad r_2 = 0.420$$

$$\tau = -\ln(r_1)/(10+0+0) = 0.535$$

$$\theta = r_2 \times (10+0+0) = 4.200$$

Iteration 1



$t = 0$

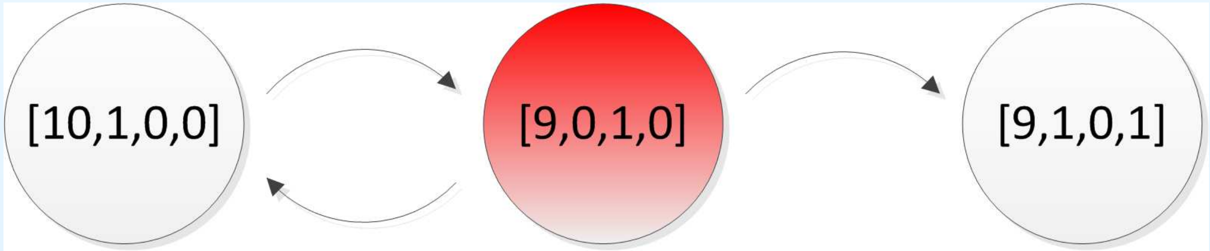
Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 0$	0
R_2	$k_2 x_C = 1$	1
R_3	$k_3 x_C = 0.01$	1.01

$$r_1 = 0.297, \quad r_2 = 0.520$$
$$\tau = -\ln(r_1)/(0+1+0.01) = 1.202$$
$$\theta = r_2 \times (0+1+0.01) = 0.525$$

Iteration 2



$$t = 0.535$$

Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

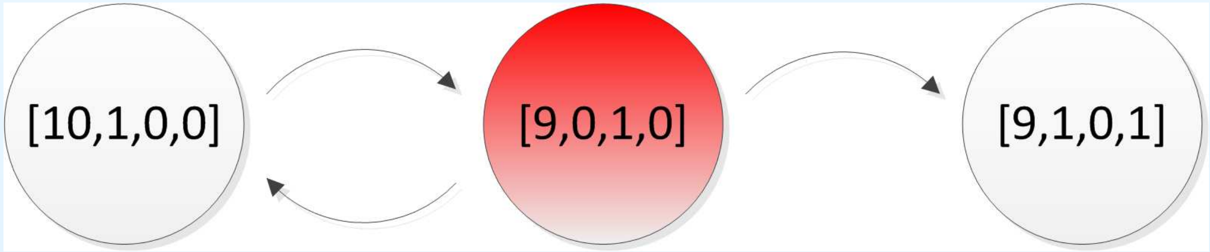
Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 0$	0
R_2	$k_2 x_C = 1$	1
R_3	$k_3 x_C = 0.01$	1.01

$$r_1 = 0.297, \quad r_2 = 0.520$$

$$\tau = -\ln(r_1)/(0 + 1 + 0.01) = 1.202$$

$$\theta = r_2 \times (0 + 1 + 0.01) = 0.525$$

Iteration 2



$$t = 0.535$$

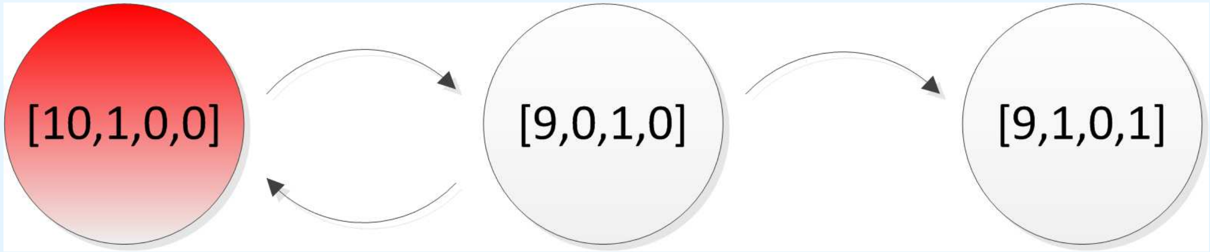
Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 10$	10
R_2	$k_2 x_C = 0$	10
R_3	$k_3 x_C = 0$	10

$$r_1 = 0.210, \quad r_2 = 0.647$$
$$\tau = -\ln(r_1)/(10+0+0) = 0.156$$
$$\theta = r_2 \times (10+0+0) = 6.47$$

Iteration 3



$$t = 1.737$$

Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

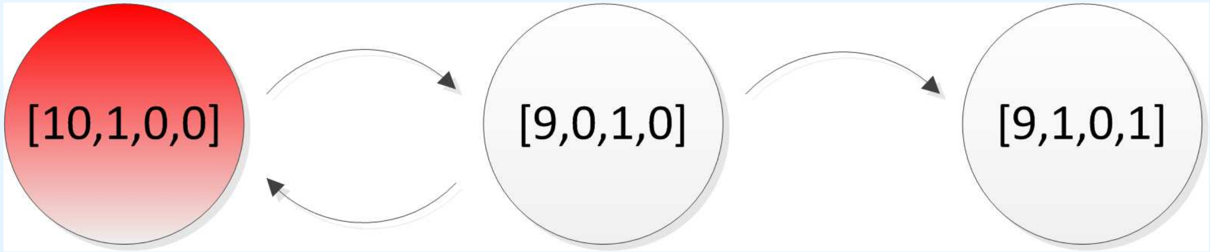
Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 10$	10
R_2	$k_2 x_C = 0$	10
R_3	$k_3 x_C = 0$	10

$$r_1 = 0.210, \quad r_2 = 0.647$$

$$\tau = -\ln(r_1)/(10+0+0) = 0.156$$

$$\theta = r_2 \times (10+0+0) = 6.47$$

Iteration 3



$$t = 1.737$$

Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

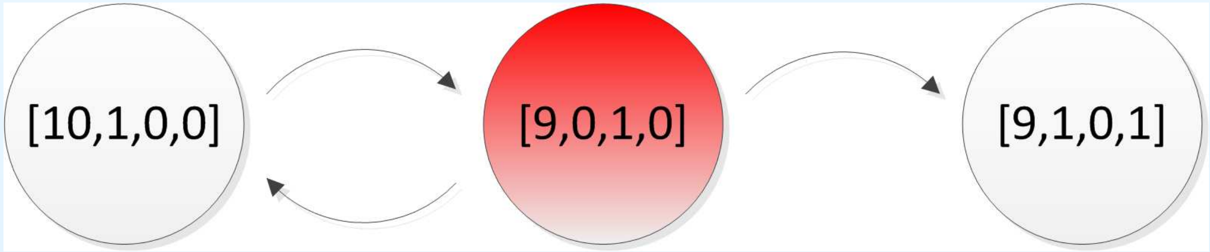
Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 0$	0
R_2	$k_2 x_C = 1$	1
R_3	$k_3 x_C = 0.01$	1.01

$$r_1 = 0.312, \quad r_2 = 0.849$$

$$\tau = -\ln(r_1)/(0+1+0.01) = 1.153$$

$$\theta = r_2 \times (0+1+0.01) = 0.857$$

Iteration 4



$$t = 1.893$$

Sample SSA Run of Enzymatic Reaction (Direct Method)

An SSA simulation run with initial condition: $\mathbf{X}(0) \equiv (X_S(0), X_E(0), X_C(0), X_P(0)) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

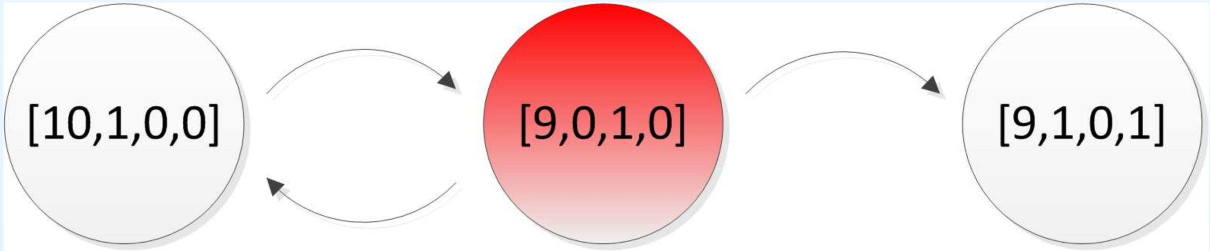
Reaction	Propensity	Partial sum
R_1	$k_1 x_S x_E = 0$	0
R_2	$k_2 x_C = 1$	1
R_3	$k_3 x_C = 0.01$	1.01

$$r_1 = 0.312, \quad r_2 = 0.849$$

$$\tau = -\ln(r_1)/(0 + 1 + 0.01) = 1.153$$

$$\theta = r_2 \times (0 + 1 + 0.01) = 0.857$$

Iteration 4



$$t = 1.893$$

Multi-Timescale Problem with SSA

An SSA simulation run with initial condition: $\mathbf{X}(0) = (10, 1, 0, 0)$, and with rate constants: $k_1 = 1,\ k_2 = 1,\ k_3 = 0.01$.

- On average, we encounter 100 dissociation reaction events before we observe the next production reaction event.
- We spend lots of CPU time for uninteresting reaction events.

More extreme case with initial condition: $\mathbf{X}(0) = (3000, 220, 0, 0)$, and with rate constants: $k_1 = 0.01,\ k_2 = 100,\ k_3 = 0.01$:

- 1,000 simulation runs of 20,000 time units took over 68 hours on a 3GHz Pentium 4 machine.

In general, when $k_2 \gg k_3$:

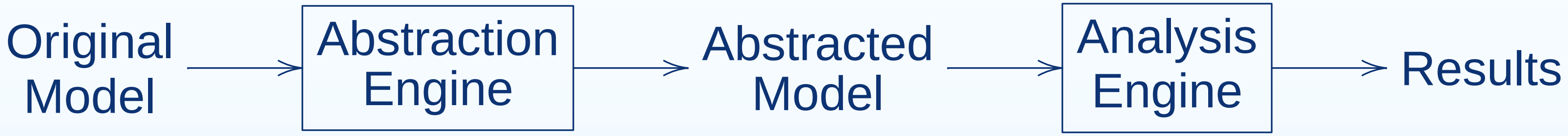
- Most of computation time is allocated for simulating formations and breakups of C.
- Very unproductive.

Bottom Line

SSA can be very expensive not only because it can require a very large number of simulation runs to obtain statistically meaningful results but also because it simulates each reaction event one at a time.

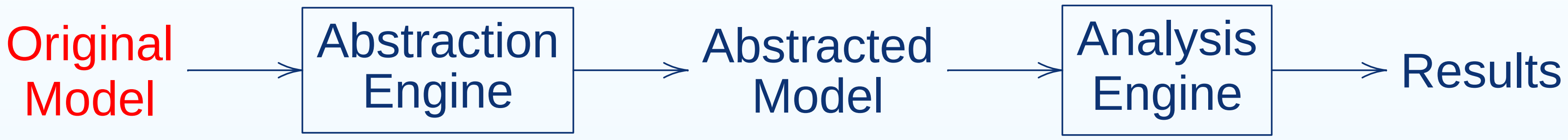
- A higher level abstraction is essential for analysis of large multiscale systems.
- Essential to balance accuracy and efficiency.
- However, it is hard to do in general setting.
- One approach is to reduce commonly seen network structures at various resolutions.

Our Automated Modeling and Analysis Tool Flow



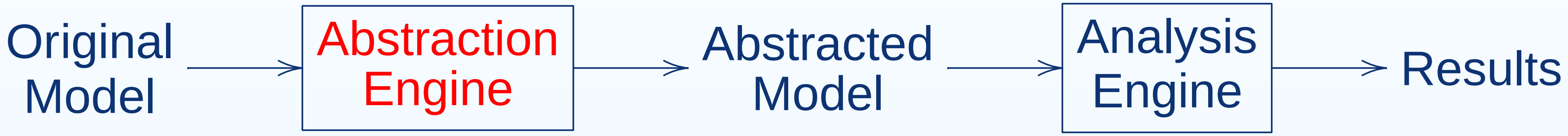
- Our approach to accelerate temporal behavior analysis.

Our Automated Modeling and Analysis Tool Flow



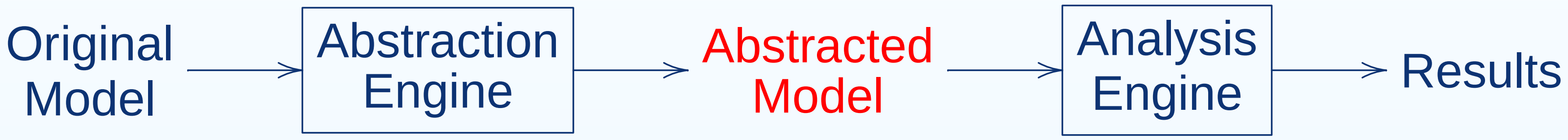
- Reaction-based model in SBML format.
- Usually a low-level abstraction (elementary reaction level).
- Requires substantial computational costs for analysis.

Our Automated Modeling and Analysis Tool Flow



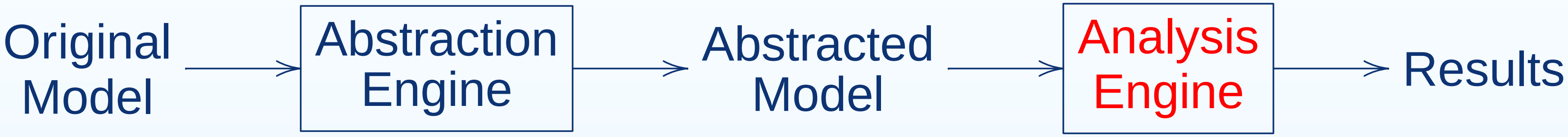
- Contains a suite of model abstraction methods.
- User can configure which methods to apply.
- Systematically checks conditions for each model abstraction.
- Automatically performs transformations.
- Faster and more accurate compared with manual model abstraction.
- Easy to generate models with various level of resolutions.

Our Automated Modeling and Analysis Tool Flow



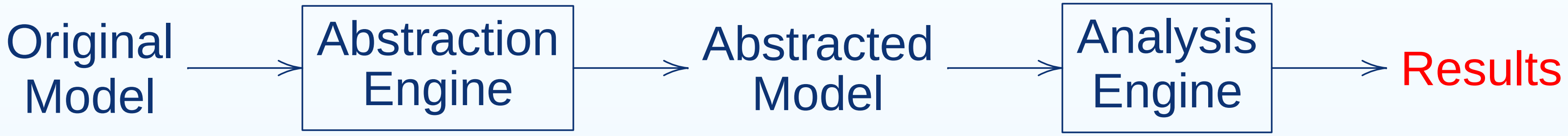
- A higher-level model which contains fewer species and reactions.
- Easier to intuitively visualize crucial components and interactions.
- Many fast reactions are removed.
- Substantially lowers the cost of stochastic analysis.
- Can be saved as SBML.

Our Automated Modeling and Analysis Tool Flow



- Various Monte Carlo simulation methods including the SSA.
- Various ODE simulation methods.
- Efficient probabilistic analysis features.

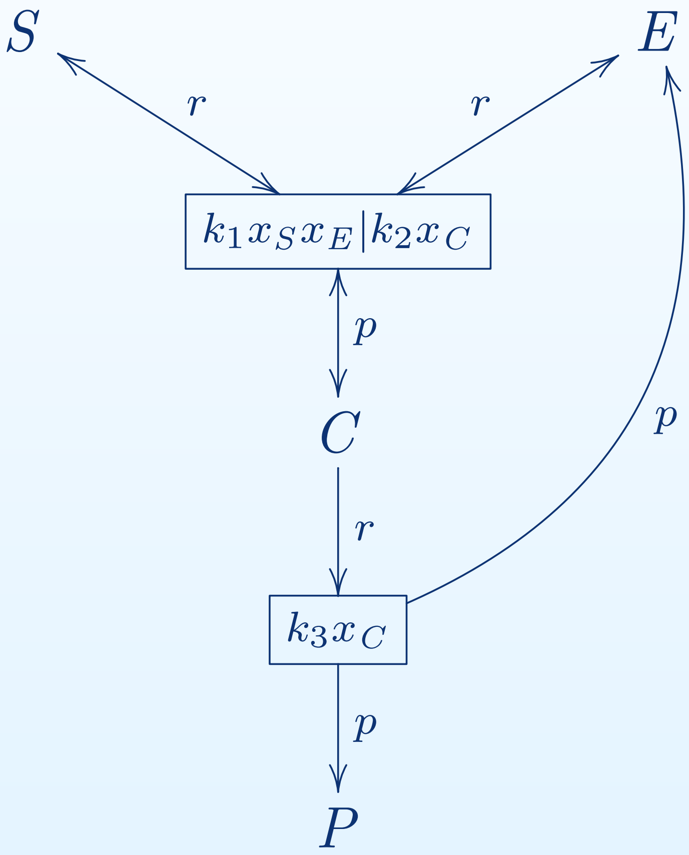
Our Automated Modeling and Analysis Tool Flow



- Can be obtained significantly faster.
- Can approximate the original model well.

Model Representation of Enzymatic Reaction

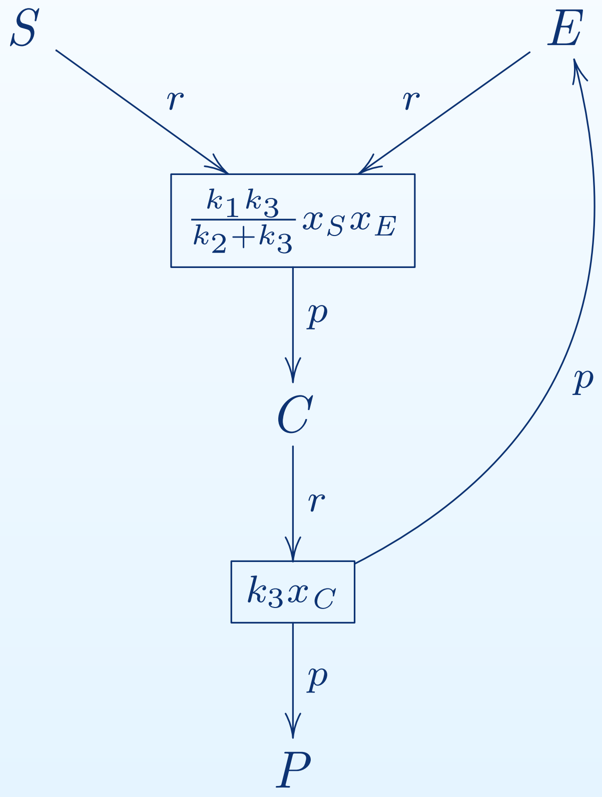
Model: $E + S \underset{k_2}{\overset{k_1}{\rightleftarrows}} C \xrightarrow{k_3} E + P$.



- Bipartite graph with species nodes and reaction nodes.
- Double arrows represent reversible reactions.
- 4 species and 3 reactions.
- Unproductive when $k_2 \gg k_3$.

Production-Passage-Time Approximation

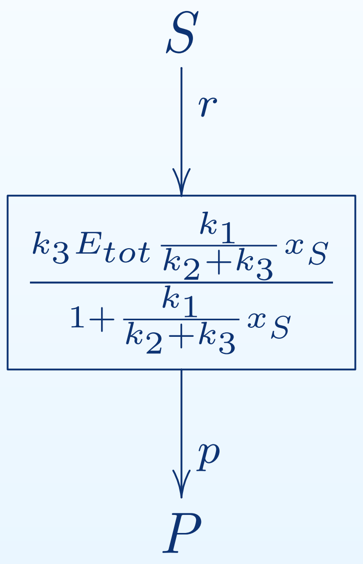
The idea: simple model reduction to minimize the number of reaction events that fire in each simulation of the enzymatic reaction.



- Removes unproductive reaction.
- Approximates passage time of C formation leading to P production.
- 4 species and 2 reactions.

Quasi-Steady-State Approximation

Assumes C in steady state, and deterministically and algebraically expresses x_C in terms of x_S.



S

r

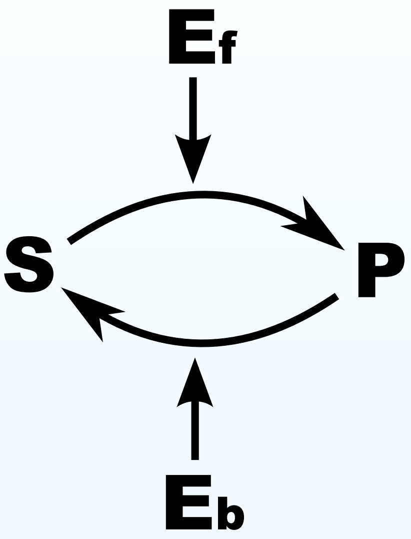
$$\boxed{\frac{k_3 E_{tot} \frac{k_1}{k_2+k_3} x_S}{1+\frac{k_1}{k_2+k_3} x_S}}$$

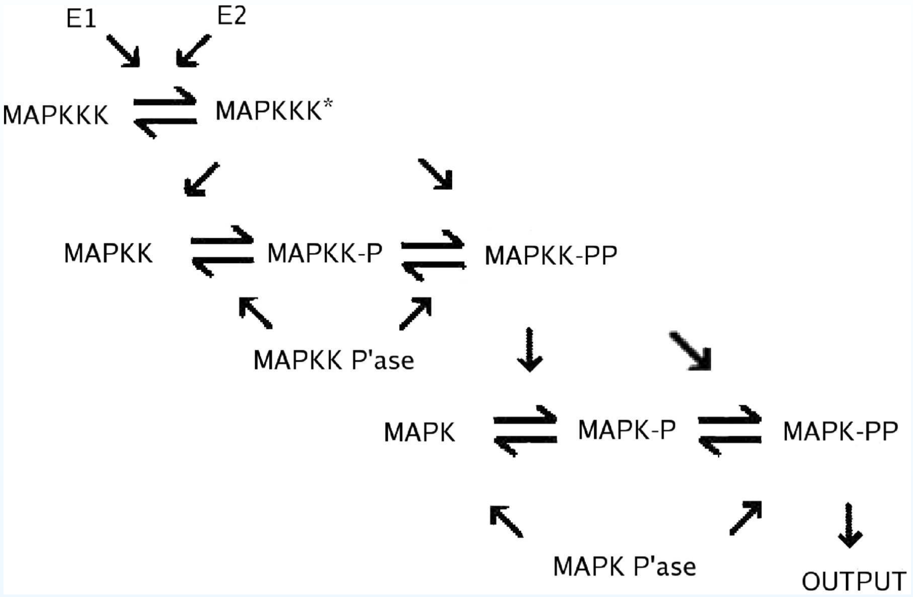
p

P

- Removes fast reactions.
- Further reduces dimensionality.
- 2 species and 1 reaction.
- $E_{tot} \ll S_{tot} + \frac{k_2+k_3}{k_1}$.

Enzymatic Cycle





- Ubiquitous control motif.
- Has two enzymatic reactions.
- Models regulation of protein activity.
- Can have rich dynamics:
  - Ultrasensitivity.
  - Adaptation.
  - Bistable oscillation.

$$E_f + S \underset{k_2}{\overset{k_1}{\rightleftarrows}} C_f \xrightarrow{k_3} E_f + P$$

$$E_b + P \underset{k_5}{\overset{k_4}{\rightleftarrows}} C_b \xrightarrow{k_6} E_b + S$$

Enzymatic Cycle Example 1

$$E_f + S \underset{k_2}{\overset{k_1}{\rightleftarrows}} C_f \xrightarrow{k_3} E_f + P, \quad E_b + P \underset{k_5}{\overset{k_4}{\rightleftarrows}} C_b \xrightarrow{k_6} E_b + S$$

with the initial conditions:

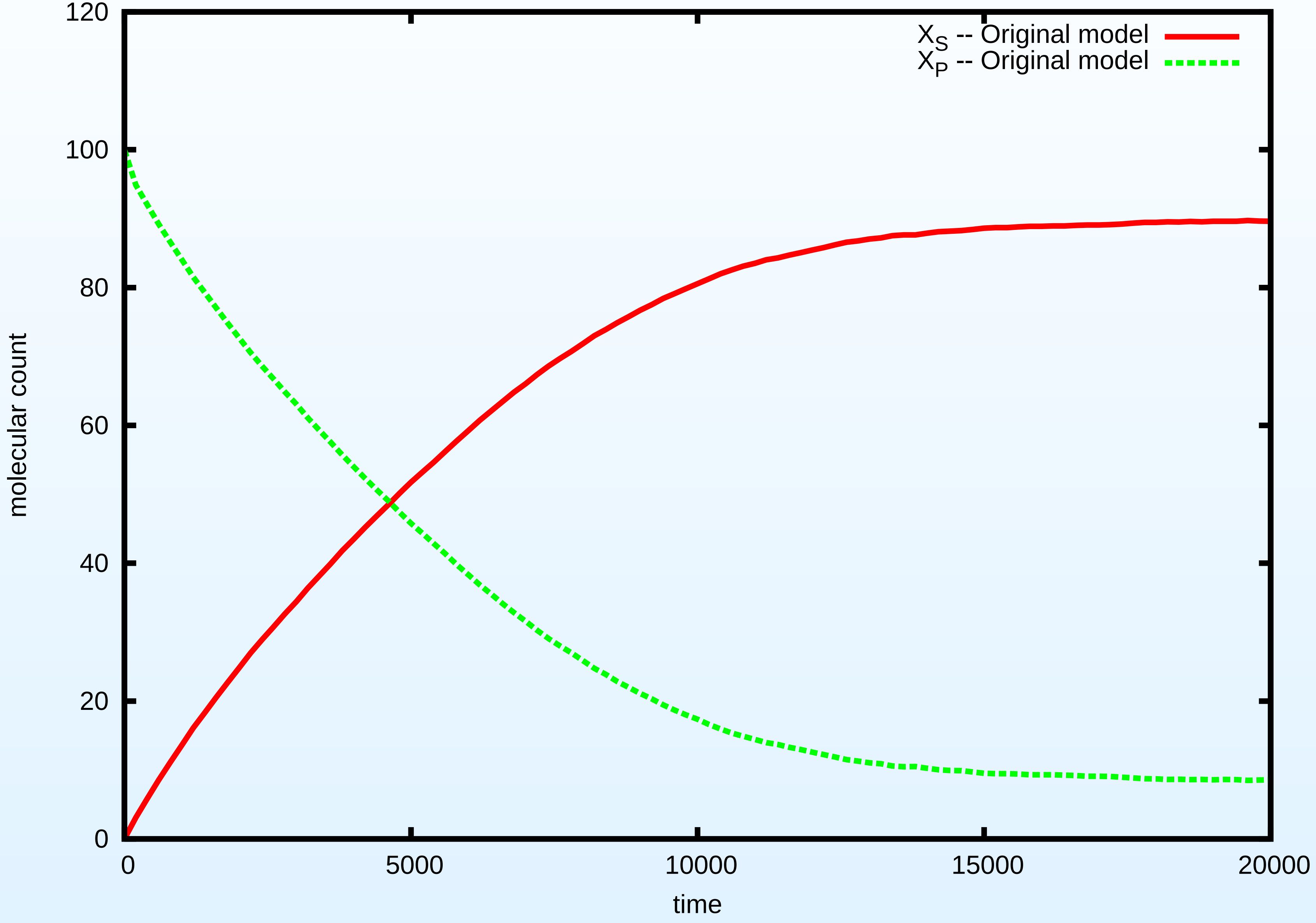
$$(X_S(0), X_P(0), X_{E_f}(0), X_{E_b}(0), X_{C_f}(0), X_{C_b}(0)) = (100, 0, 2, 1, 0, 0).$$

The rate constants:

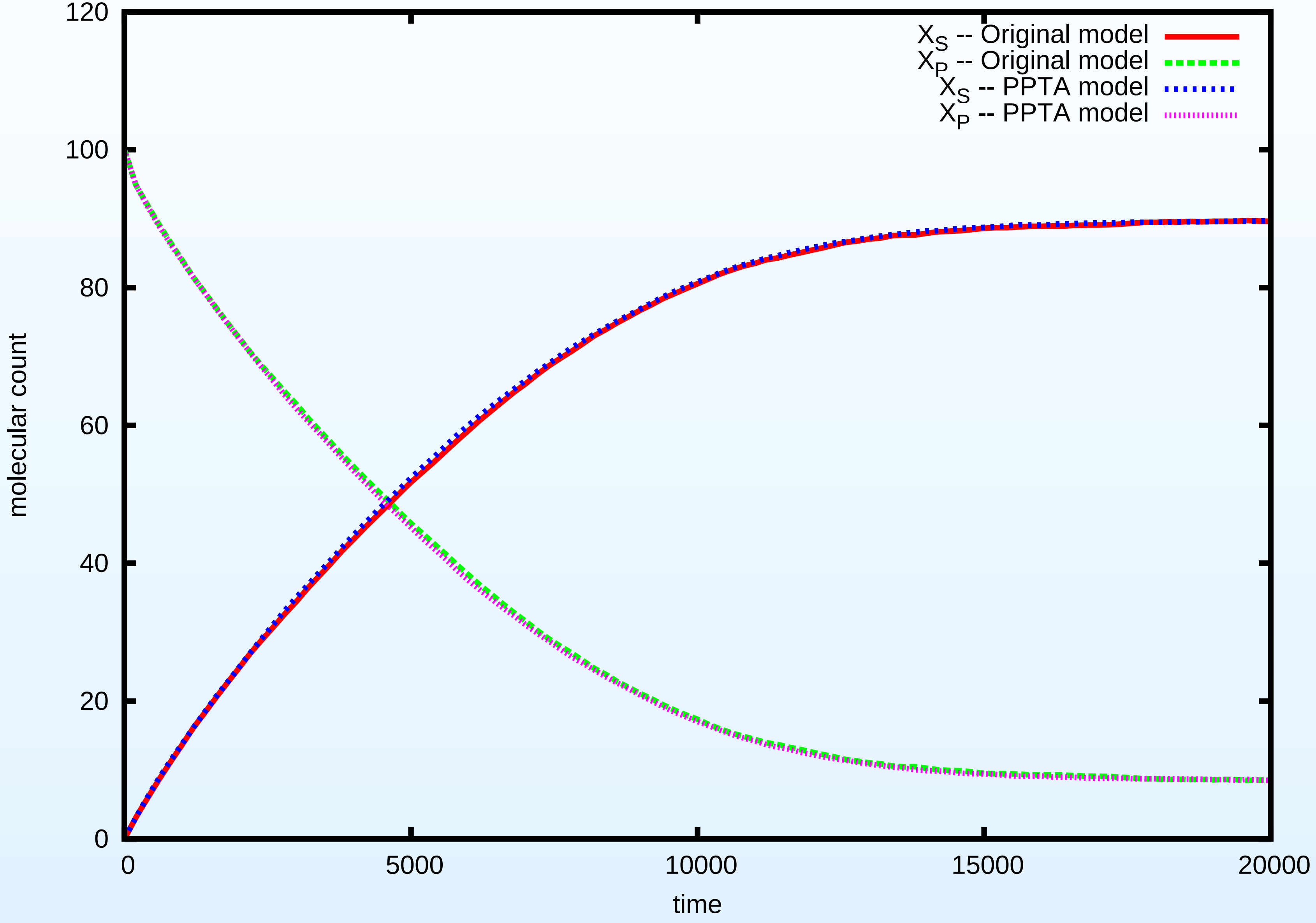
$$k_1 = 0.1; k_2 = 1.0; k_3 = 0.01; k_4 = 0.1; k_5 = 1.0; \text{ and } k_6 = 0.01.$$

- Run for 20000 time units.
- Simulated for 1,000 runs.

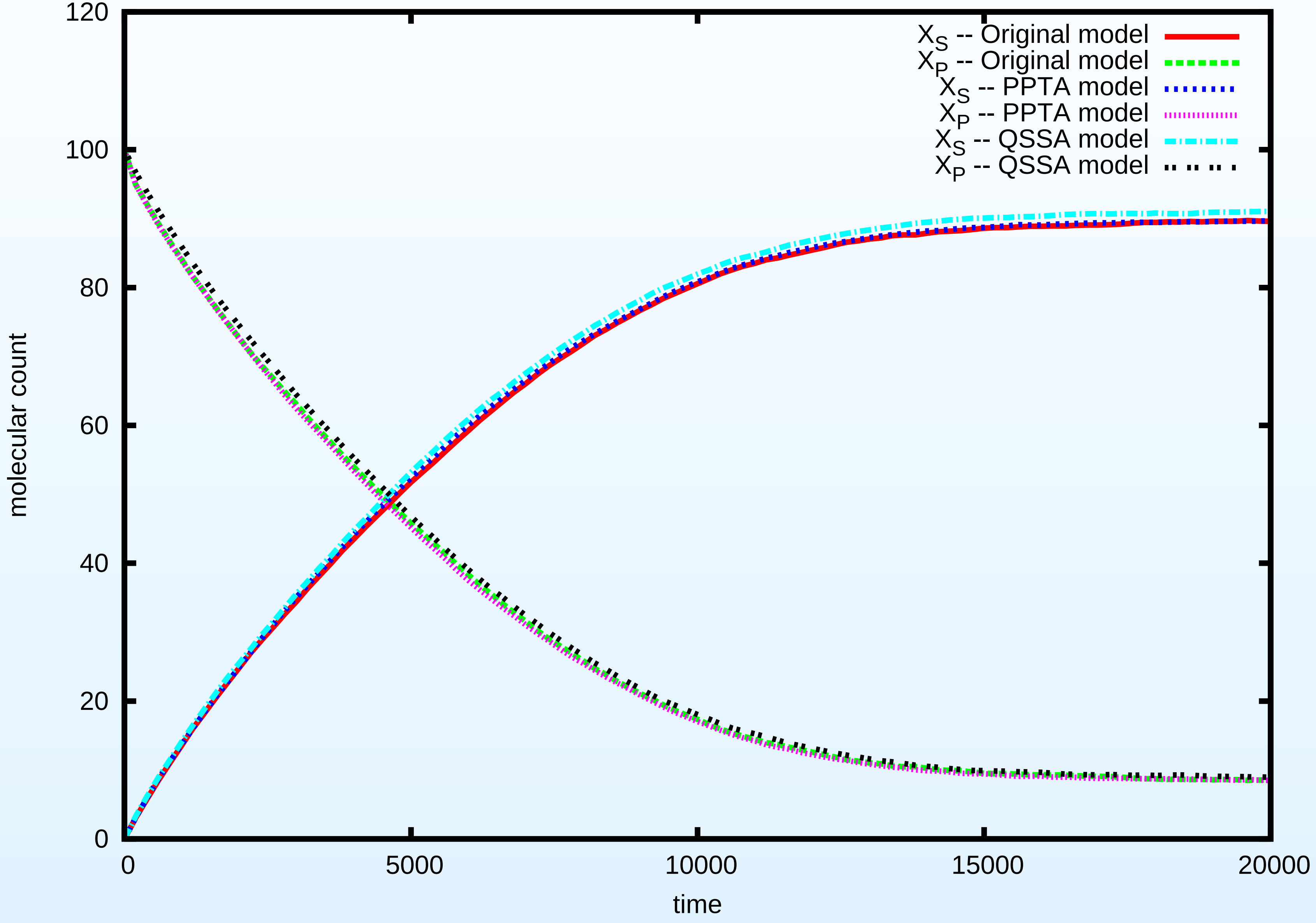
Enzymatic Cycle Example 1: Accuracy



Enzymatic Cycle Example 1: Accuracy



Enzymatic Cycle Example 1: Accuracy



Enzymatic Cycle Example 1: Efficiency

Model	Time	Speedup
Original	228s	1
PPTA	17s	13
QSSA	12s	19

Enzymatic Cycle Example 2

$$E_f + S \underset{k_2}{\overset{k_1}{\rightleftarrows}} C_f \xrightarrow{k_3} E_f + P, \quad E_b + P \underset{k_5}{\overset{k_4}{\rightleftarrows}} C_b \xrightarrow{k_6} E_b + S$$

with the initial conditions:

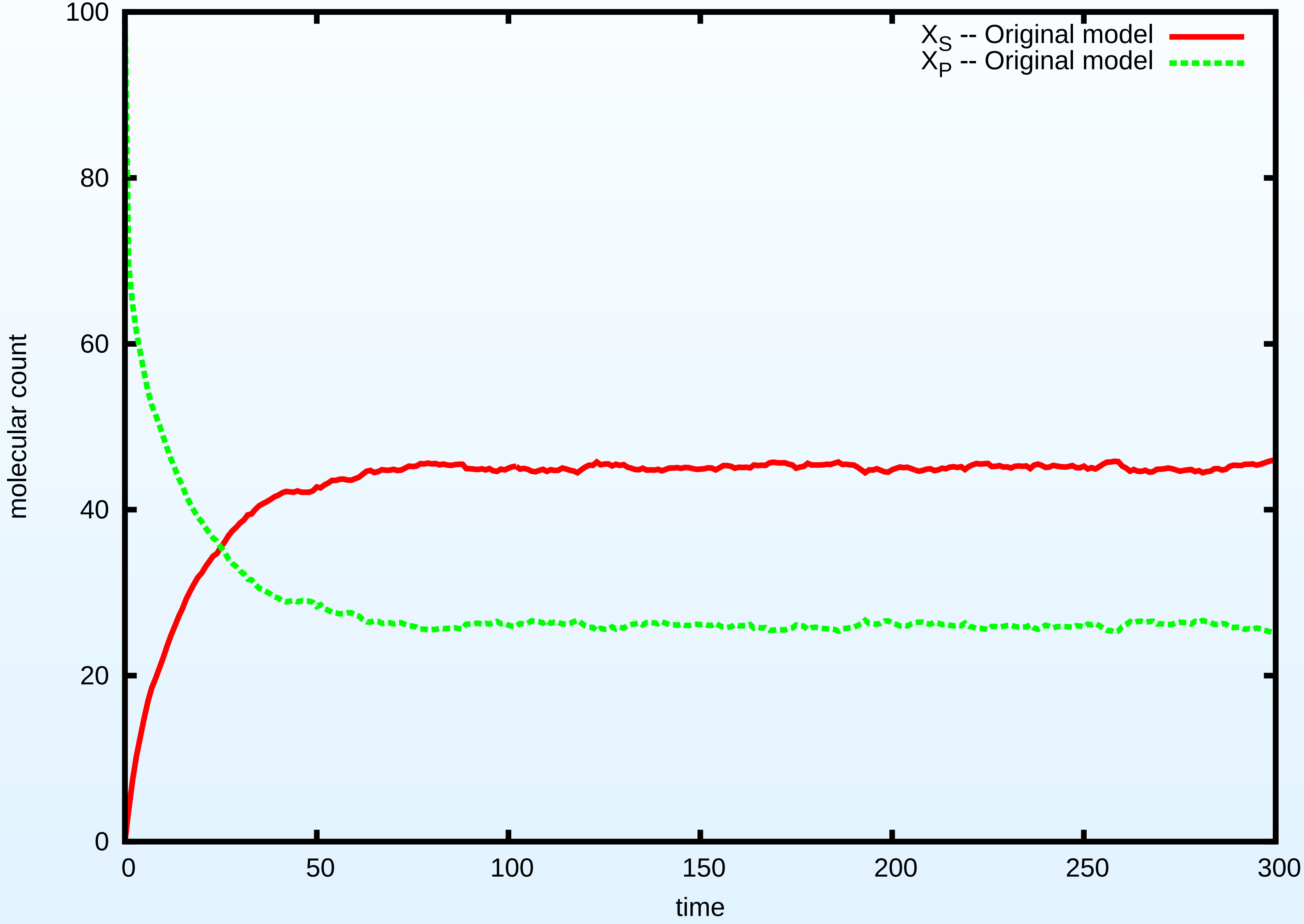
$$(X_S(0), X_P(0), X_{E_f}(0), X_{E_b}(0), X_{C_f}(0), X_{C_b}(0)) = (0, 100, 10, 20, 0, 0).$$

The rate constants:

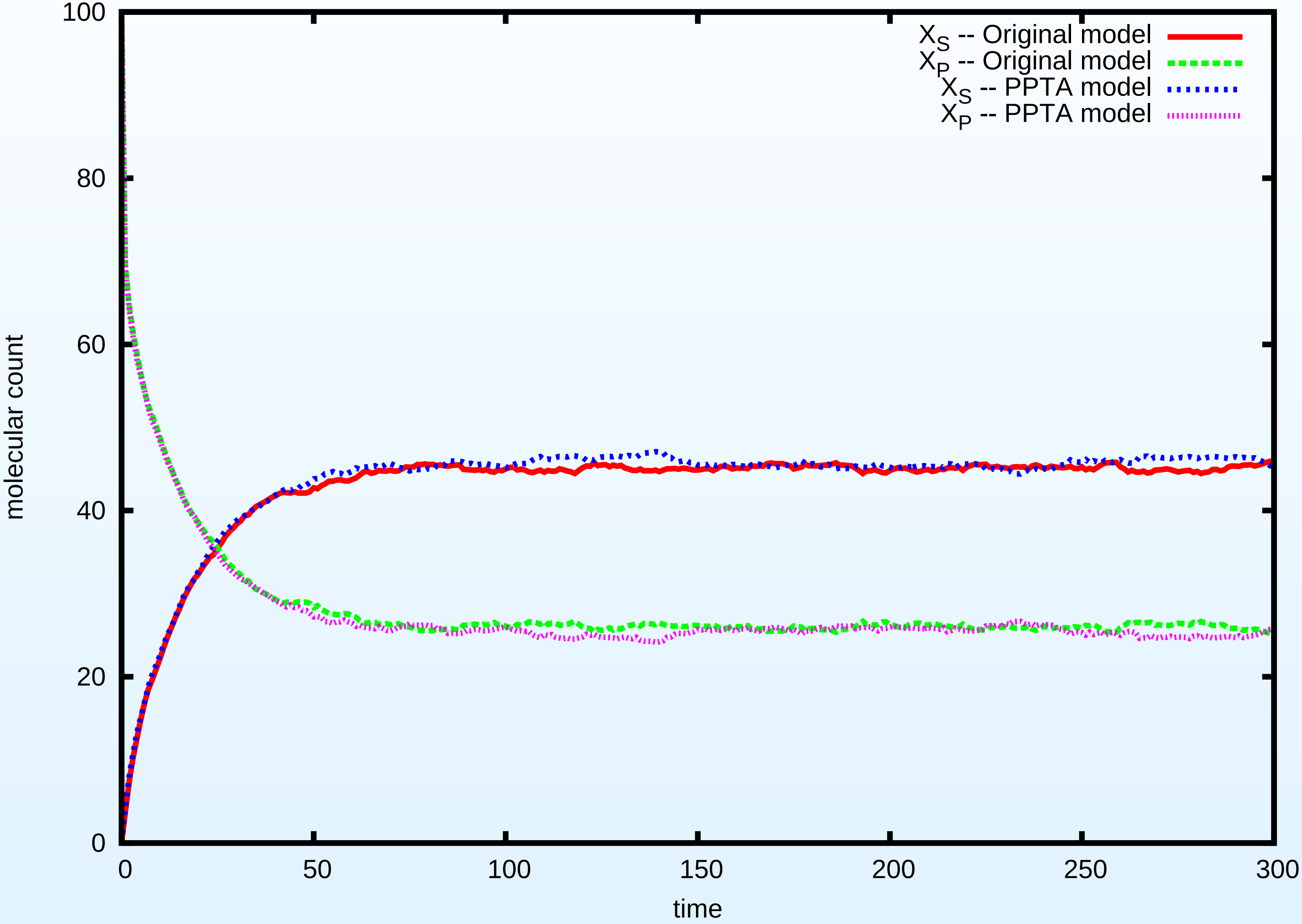
$$k_1 = 10^3; k_2 = 1.5 \times 10^3; k_3 = 2; k_4 = 10^3; k_5 = 5 \times 10^2; \text{ and } k_6 = 1.$$

- Run for 300 time units.
- Simulated for 1,000 runs.

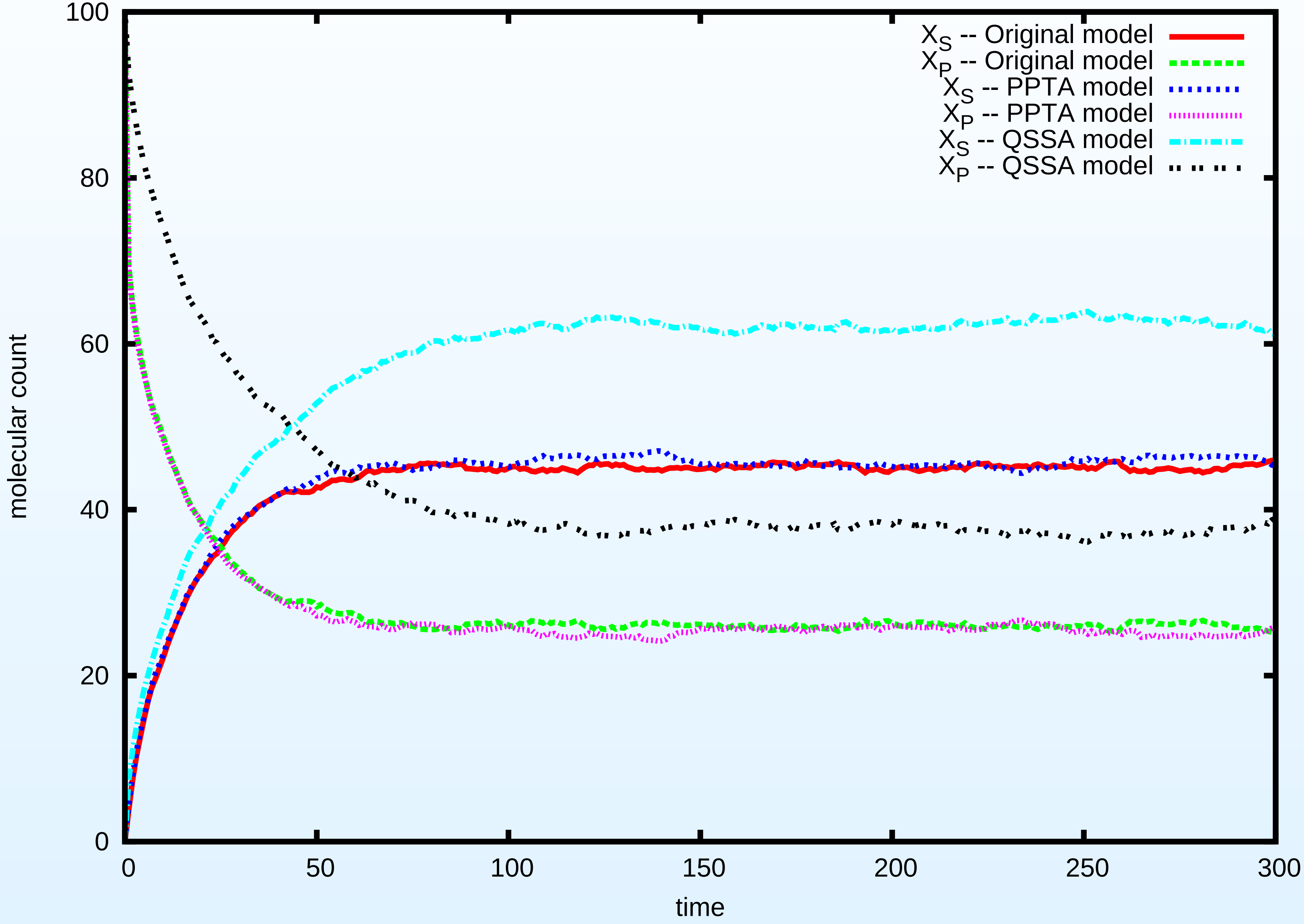
Enzymatic Cycle Example 2: Accuracy



Enzymatic Cycle Example 2: Accuracy



Enzymatic Cycle Example 2: Accuracy



Enzymatic Cycle Example 2: Efficiency

Model	Time	Speedup
Original	17.73h	1
PPTA	87.51s	729
QSSA	53.43s	1,194

Rare yet Catastrophic Events

- Natural biological systems are robust to a certain range of internal and external variations.
- Occurrence of failure events may be rare under normal settings.
- However, when they happen, they can lead to catastrophic consequences.
- By treating complex non-Mendelian diseases as system failure, *in silico* rare event analysis can become an important tool to understand disease etiology.
- Rare event analysis presents a particularly challenging computational problem.

```
                         WARNING!

The system is either busy or has become unstable. You can wait and
see if it becomes available again, or you can restart your computer.

* Press any key to return to Windows and wait.
* Press CTRL+ALT+DEL again to restart your computer. You will
  lose unsaved information in any programs that are running.

                 Press any key to continue _
```

Transition Event Analysis via Simulation

Objective: Estimate $p \equiv P_{t \leq t_{\max}}(\mathbf{X} \rightarrow \mathcal{E} \mid \mathbf{x_0})$, the probability that $\mathbf{X}$ moves to any states in $\mathcal{E}$ within $t_{\max}$ given $\mathbf{X}(0) = \mathbf{x_0}$.

- Define Y be a Boolean random variable such that:

$$Y = \begin{cases} 1 & \text{if the system moves to } \mathcal{E} \text{ within } t_{\max}, \\ 0 & \text{otherwise.} \end{cases}$$

- Also, let $Y^{\{i\}}$ be the i-th sample of Y. Then generate n samples of Y by running n simulation of $\mathbf{X}(t)$, and estimate p by p_n:

$$p_n \equiv \frac{1}{n} \sum_{i=1}^{n} Y^{\{i\}}.$$

Problem with This Approach

Since we only use 0 and 1, it takes very large n to estimate very small p.

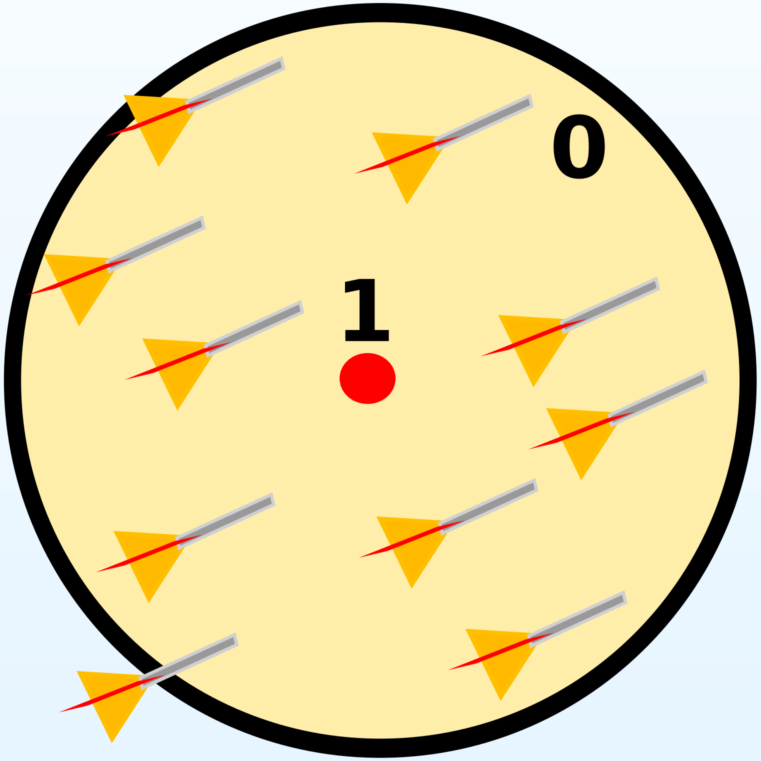
For example, suppose $p = 10^{-6}$:

- On average, it takes 10^6 samples to get the first hit.
- With $n = 10^5$, $p_n = 10^{-5}$ with one hit, $p_n = 0$ with no hit.
- Very sensitive to 1’s.
- Has high variance.

Importance Sampling

Instead of using rare 1’s for hits, use much more frequent smaller number.

Suppose $p = 0.005$.

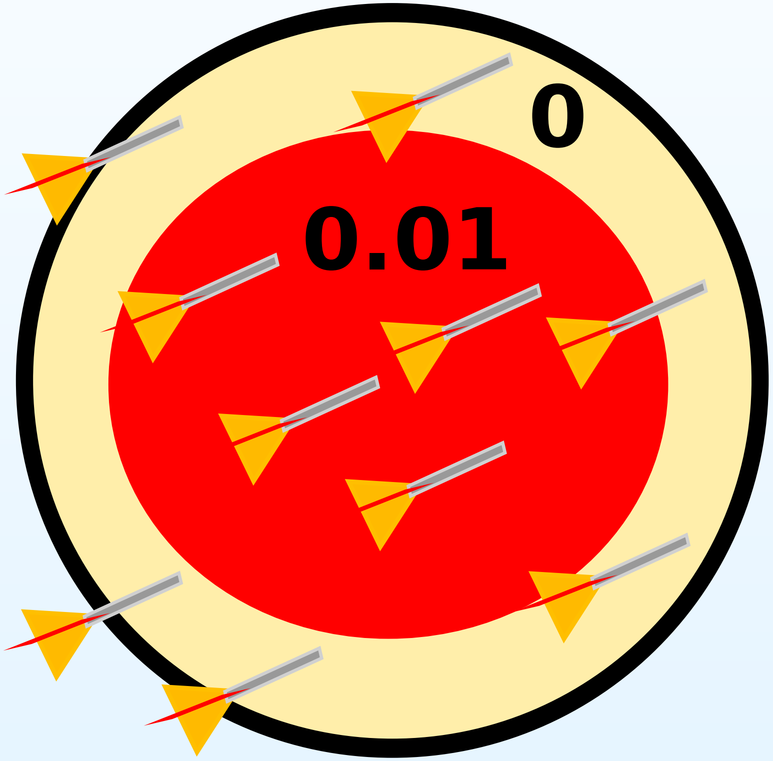


$$p_{10} = 0/10 = 0$$

Importance Sampling

Instead of using rare 1's for hits, use much more frequent smaller number.

Suppose $p = 0.005$.



$$p_{10} = 0.04/10 = 0.004$$

Weighted Stochastic Simulation Algorithm (wSSA)

Idea: bias reaction selection to observe $\mathbf{X} \to \mathcal{E}$ more often and weight each outcome to correct the sampling bias.

- Next reaction is selected using biased propensity functions $b_j(\mathbf{x})$:

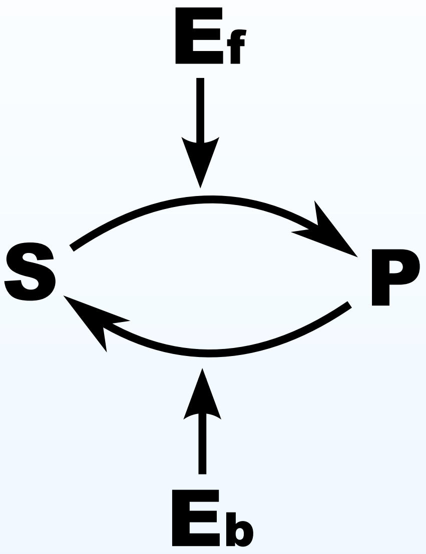
$$Prob(j \mid \mathbf{x}) = \frac{b_j(\mathbf{x})}{\sum_{j'} b_{j'}(\mathbf{x})}.$$

- To compensate this bias in the reaction selection, the weight factor

$$w(j;\mathbf{x}) = \frac{a_j(\mathbf{x}) \sum_{j'=1}^{M} b_{j'}(\mathbf{x})}{b_j(\mathbf{x}) \sum_{j'=1}^{M} a_{j'}(\mathbf{x})}$$

  is used to reflect the likelihood of the reaction selection.
- Each run has a weight based on the product of all $w(j;\mathbf{x})$.
- Each weight is usually less than 1, so we can have smaller variance.

Rare Event Analysis: Balanced Enzymatic Cycle



$$E_f + S \underset{k_2}{\overset{k_1}{\rightleftarrows}} C_f \xrightarrow{k_3} E_f + P,\ E_b + P \underset{k_5}{\overset{k_4}{\rightleftarrows}} C_b \xrightarrow{k_6} E_b + S$$

$$X_{E_*}(0) = 1; X_S(0) = X_P(0) = 50; X_{C_*}(0) = 0,$$

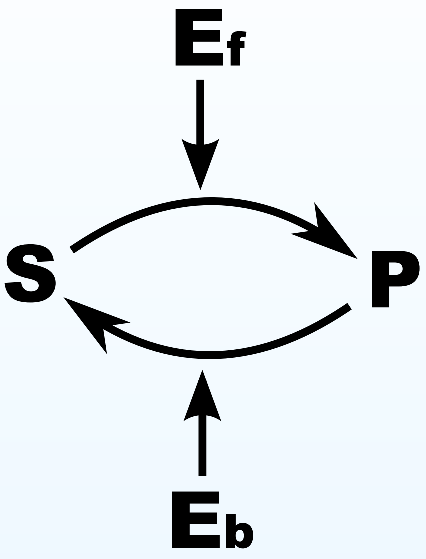
$$k_1 = k_2 = k_4 = k_5 = 1; k_3 = k_6 = 0.1.$$

With this condition, X_S and X_P typically stay around 50.

We are interested in estimating the probability that X_P moves to 25 within 100 time units. The true probability is:

$$P_{t \le 100}(X_P \rightarrow 25 \mid \mathbf{x_0}) = 1.738153 \times 10^{-7}.$$

wSSA Rare Event Analysis: Balanced Enzymatic Cycle



$$E_f + S \underset{k_2}{\overset{k_1}{\rightleftarrows}} C_f \xrightarrow{k_3} E_f + P,\ E_b + P \underset{k_5}{\overset{k_4}{\rightleftarrows}} C_b \xrightarrow{k_6} E_b + S$$

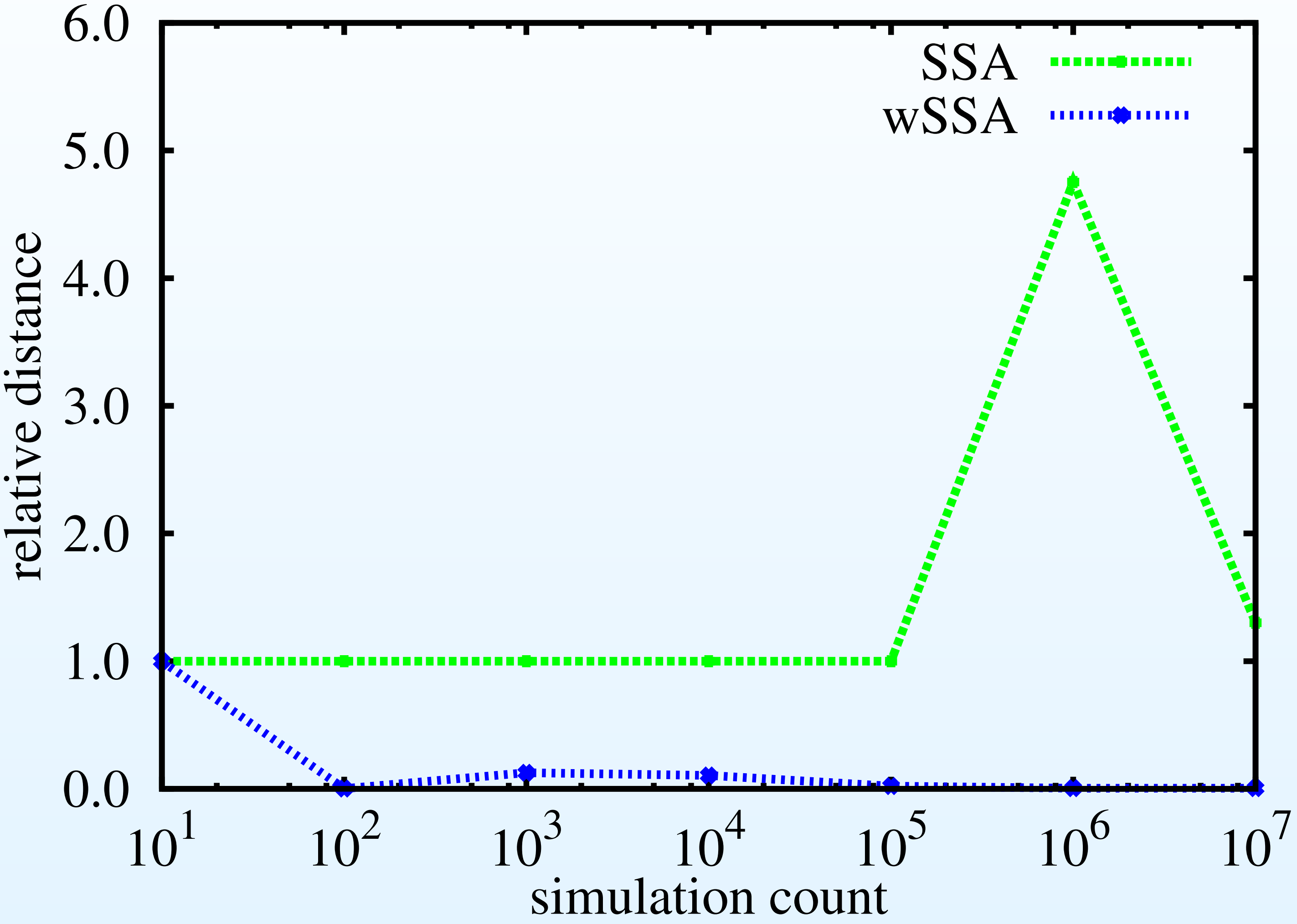
$$X_{E_*}(0) = 1; X_S(0) = X_P(0) = 50; X_{C_*}(0) = 0,$$

$$k_1 = k_2 = k_4 = k_5 = 1; k_3 = k_6 = 0.1.$$

In order to observe $X_P \to 25$ more often, the following biased propensity functions are used:

$$b_3(\mathbf{x}) = 0.5 \times a_3(\mathbf{x}),$$
$$b_6(\mathbf{x}) = 2.0 \times a_6(\mathbf{x}).$$

Balanced Enzymatic Cycle Results



Conclusions

- Stochastic simulation becomes an important tool to study stochastic effects on system-level properties.
- Stochastic simulation can be very expensive.
- Modeling and analysis method should be tailored for specific properties of interest.
- For multiscale system, model abstraction can be useful.
- For rare event analysis, wSSA can be useful.

Acknowledgment

- Chris J. Myers (University of Utah)
- Michael Samoilov (QB3: UCB – California Institute for Quantitative Biosciences)
- Ivan Mura (University of Trento – Microsoft Research CoSBi)