Efficient Finite Difference Method for Computing Sensitivities of Biochemical Reactions

Vo Hong Thanh\textsuperscript{a,}, Roberto Zunino\textsuperscript{b}, Corrado Priami\textsuperscript{a,b}

\textsuperscript{a}The Microsoft Research – University of Trento Centre for Computational and Systems Biology (COSBI)
\textsuperscript{b}University of Trento, Department of Mathematics

Abstract

Sensitivity analysis of biochemical reactions aims at quantifying the dependence of the reaction dynamics on the reaction rates. The computation of the parameter sensitivities poses many computational challenges when taking stochastic noise into account. This paper proposes a new efficient finite difference method for computing parameter sensitivities of biochemical reactions. We employ propensity bounds of reactions to simulate the nominal and perturbed processes and build the estimator. The exactness of the simulation is reserved by applying the rejection-based mechanism. Our approach reduces the variance of the estimator by exploiting the positive correlation of these processes and improves the performance by skipping the propensity updates. Furthermore, by using propensity bounds to couple processes, our approach allows simultaneous perturbation of many reaction rates in computing the sensitivities, which further improves the efficiency of the estimator. We benchmark our method on reaction models to prove its applicability and efficiency.

Keywords: Computational biology, Rejection-based stochastic simulation, Parameter sensitivity, Rejection-based finite difference method

1. Introduction

Biochemical reactions at cellular level are an inherently nonlinear and stochastic due to the discreteness in the copy numbers of molecular species and randomness in molecular collisions enabling the reactions between these species. The stochasticity of biochemical reactions (often referred to as biological noise) is further amplified when key species like genes, mRNA are often present with low copy numbers. The effects of biological noise have been demonstrated to play an important role in driving biological processes like gene regulation\cite{1,2,3,4,5} or cell fate decision\cite{6}. The noise may be further propagated across cells leading to remarkable diversity at organism level\cite{7,8}.

\textsuperscript{*}Corresponding author

Email addresses: vo@cosbi.eu (Vo Hong Thanh), roberto.zunino@unitn.it (Roberto Zunino), priami@cosbi.eu (Corrado Priami)
Stochastic chemical kinetics has been adopted to study dynamical behavior of biochemical reactions where stochastic noise is treated as an intrinsic part. It acknowledges the discrete nature of molecular species by keeping track of the discrete copy number of each species, called population. The collection of populations of species forms the system state. The possibility that a reaction occurs in the next time is assigned a probability which is proportional to a propensity function. The propensity of a reaction depends on the reactant species and on its reaction rate. The probability distribution of the system state over time is characterized by the chemical master equation (CME) \[9\] and can be exactly realized by the Gillespie’s stochastic simulation algorithm (SSA) \[10, 11\]. The core of SSA is a Monte Carlo procedure that moves the system state by randomly selecting a reaction to fire according to its propensity. Two first implementations of the Monte Carlo step are the direct method (DM) and first reaction method (FRM) \[10\]. Since then, many efficient implementations of the Monte Carlo step have been introduced including the next reaction method (NRM) \[12, 13\], DM with sorted reactions \[14, 15\], DM with tree-based search \[16, 17, 18, 19, 20\], DM with composition-rejection search (DM-CR) \[21, 22\], the partial-propensity DM (PDM) \[24, 23\], the rejection-based SSA (RSSA) \[25, 26, 27, 28\] and others \[29, 30, 31\]. The extensions of SSA have been introduced to cope with different aspects of biochemical reactions like time delays \[32, 33, 34, 13, 35\] and time-dependent reaction rates \[36, 13, 37\]. In addition, approximate algorithms which improve the computational performance by sacrificing the simulation correctness have been also developed \[38, 39, 40, 41, 42, 43\].

The dynamical behavior of biochemical reaction systems is affected by reaction rates. The changes in the rates of reactions, hence reaction propensities, due to, for example, changes in the cellular environment and/or measurement errors may significantly alter the system behavior. Therefore it is important to quantify the dependence of the reaction system dynamics on reaction rates. Sensitivity analysis aims at quantitatively characterizing this dependency. Different methods for sensitivity analysis of stochastic chemical kinetics have been introduced including finite differences \[44, 45, 46, 48\], likelihood ratios \[49, 50, 47\] and infinitesimal perturbation analysis \[51, 52\]. Each of these methods has its own advantages and drawbacks (see Asmussen and Glynn \[53\] for a general discussion on these methods).

The finite difference scheme measures the difference in the behavior of reactions by imposing a small perturbation to reaction rates around their nominal values. A crude estimator for computing sensitivities based on the finite difference approach is to use two independent SSA simulation runs in which the first one simulates reactions with nominal rate values and the second one applies to perturbed values, respectively. The crude estimator requires two streams of independent random numbers and often results in large variance. To reduce the variance of the estimator, Rathinam et al. \[44\] introduced the common random number (CRN) method where the same stream of random numbers is used for both the simulations of reactions with nominal and perturbed rates. However, the correlation of the nominal and perturbed processes introduced by CRN during the simulation is broken if the simulation time is large. Anderson \[45\] introduced the coupled finite difference (CFD) which improves the estimator by coupling the nominal and perturbed processes by exploiting the random time change representation in which the number of firings of reactions is modeled as unit-rate Poisson pro-
cesses with integrated propensities. For each reaction, CFD splits the Poisson processes associated with the reaction in the nominal and perturbed processes so that a common Poisson process will be shared by these processes. An additional computational cost required for sampling the shared Poisson processes is introduced into the simulation. This extra computational cost is increasing with the number of reactions and negatively affects the performance of CFD when applying for large models. We also remark that for CRN and CFD, if the sensitivities with respect to several reaction rates are required, the computation must be performed for each rate separately.

In this paper, we propose a new finite difference scheme for estimating the sensitivities of biochemical reactions. Our estimator is constructed by employing propensity bounds of reactions and the rejection-based simulation approach by Thanh et al. [25] to couple the simulation of the nominal and perturbed processes. The bounds in the propensity of a reaction are derived by bounding the population of reactant species and its rate. By using propensity bounds of reactions, we can correlate and simulate reactions with nominal and perturbed rates together, hence reducing the variance of the estimator. The exactness of the marginal distributions of the nominal and perturbed processes is preserved by applying the rejection-based mechanism. The performance gain of our approach is achieved by reducing the number of propensity updates during the simulation. Furthermore, our method can compute sensitivities by simultaneously perturbing many reaction rates in a run instead of one at a time as in CRN [44] and CFD [45]. The performance of our method is thus further improved.

The paper is organized as follows. Section 2 reviews the background of the stochastic kinetics and sensitivity analysis of biochemical reactions using stochastic simulation. Section 3 presents our new finite difference method for computing sensitivities by using the rejection-based approach. We recall the background of the rejection-based stochastic simulation algorithm that uses the concept of propensity bounds to select the next reaction firing. Then, we describe in detail our new approach to correlate the simulation of the nominal and perturbed processes employing propensity bounds of reactions in order to perform sensitivity analysis. Section 4 presents the experimental results of the application of our algorithm to actual models considered as benchmarks. The concluding remarks are in section 5.

2. Stochastic chemical kinetics

We consider a well-mixed reactor volume consisting of \( n \) molecular species represented by \( S_i \) for \( i = 1 \ldots n \). The exact population of each species \( S_i \) at a time \( t \) is kept track and denoted by \( X_i(t) \). The collection of population of species at time \( t \) forms the system state and is expressed by a \( n \)-vector \( X(t) = (X_1(t), \ldots, X_n(t)) \).

Species in the reactor volume can interact with other species through \( m \) reactions. A reaction \( R_j \) for \( j = 1 \ldots m \) describes a possible combination of species in a unidirectional way to produce other species.

\[
R_j : v_{1j}S_1 + \ldots + v_{nj}S_n \xrightarrow{\epsilon_j} v'_{1j}S_1 + \ldots + v'_{nj}S_n
\]

(1)

where the species on the left side of the arrow are called reactants and the ones on the right side are called products. A species that appears in both side of a reaction is called...
a catalyst. The non-negative integer $v_{ij}$ and $v'_{ij}$, called stoichiometric coefficients, denote the number of molecules of a reactant consumed and the number of a product produced by firing $R_j$, respectively. Each reaction is associated with a parameter $c_j$ which is called the (stochastic) reaction rate.

Each reaction $R_j$ in the stochastic chemical kinetics framework is quantified by two quantities: a state change vector $v_j$ and a propensity function $a_j$. The state change vector $v_j$ of a reaction $R_j$ is a $n$-vector where the $i$th element is $v'_{ij} - v_{ij}$. The reaction propensity $a_j$ quantifies the likeliness that a reaction $R_j$ occurs per unit time [10]. Specifically, the probability that a reaction $R_j$ fires in the next infinitesimal time $t + dt$ is $a_j(X(t))dt$, given the current state $X(t)$ at time $t$. An exact form of the propensity function $a_j$ is dependent on reaction kinetics applied. For standard mass-action kinetics, propensity $a_j$ of reaction $R_j$ is proportional its reactants and reaction rate $c_j$ and can be computed as:

$$a_j(X(t)) = c_j h_j(X(t))$$

where $h_j(X(t))$ counts the number of distinct combinations of reactants involved in $R_j$. We note that for synthesis reactions, where species are produced from an external reservoir, the number of combinations of reactants is $h_j(X(t)) = 1$.

The dynamical behavior of biochemical reactions under the stochastic chemical kinetics framework is modeled as a (continuous-time) jump Markov process where the probability distribution of all reachable states at a time is described by the chemical master equation (CME) [9]. The exact stochastic simulation algorithm (SSA) [10, 11] can be applied to construct realizations of CME. SSA is an exact algorithm in the sense that it does not introduce approximation in the sampling. The mathematical background for the simulation of SSA is the joint probability density function (pdf) $p(\tau, \mu)$ such that $p(\tau, \mu)d\tau$ gives the probability that a reaction $R_\mu$ fires in the next infinitesimal time $t + \tau + d\tau$, given the state $X(t)$ at time $t$. Its closed form is:

$$p(\tau, \mu) = a_\mu \exp(-a_0 \tau)$$

where $a_0 = \sum_{j=1}^{m} a_j$.

Various Monte Carlo approaches have been introduced for SSA in order to sample the pdf $p(\tau, \mu)$ [10, 12, 25]. One of such is the direct method (DM) which samples the pdf $p(\tau, \mu)$ in Eq. 3 by using the fact that reaction $R_\mu$ fires with a discrete probability $a_\mu/a_0$ and the firing time $\tau$ is exponentially distributed with rate $a_0$. Thus, for each simulation iteration, $m$ propensities $a_j$ for $j = 1 \ldots m$ and their sum $a_0 = \sum_{j=1}^{m} a_j$ are computed. The next reaction firing $R_\mu$ with probability $a_\mu/a_0$ is selected by

$$\mu = \text{smallest reaction index such that: } \sum_{j=1}^{\mu} a_j \geq r_1 a_0$$

and the firing time $\tau$ is generated as

$$\tau = \frac{1}{a_0} \ln \left( \frac{1}{r_1} \right)$$
where \( r_1 \) and \( r_2 \) are two random numbers generated from a uniform distribution \( U(0, 1) \). The state is moved to a new state \( X(t + \tau) = X(t) + v_\mu \). Propensities are updated as well to reflect the change in the system state. In practice, a reaction dependency graph \([13]\) is often employed to reduce the number of reaction propensities. The simulation is repeated to form a simulation trajectory until a specified ending time is reached.

2.1. Sensitivity analysis of stochastic chemical kinetics

This section deals with sensitivity analysis using stochastic simulation. We recall two finite difference schemes proposed recently for computing sensitivities: the common random number (CRN) \([44]\) and coupled finite difference (CFD) \([45]\). These methods derive the sensitivity of the system dynamics by applying a small perturbation to a reaction rate. The computation of sensitivities with respect to several reaction rates requires the computation being performed for each rate separately.

Let \( c \) be a \( m \)-vector in which the \( j \)th element is the reaction rate \( c_j \) of a reaction \( R_j \) for \( j = 1, \ldots, m \). We denote the system state at time \( t \) corresponding to rate vector \( c \) with \( X^c(t) \). Let \( f \) be a function of the state which represents a measurement of interest. The quantity \( S(c) \) that we want to measure is defined as:

\[
S(c) = \mathbb{E}[f(X^c(t))]
\]

where \( \mathbb{E}[-] \) denotes the expectation operator.

Let \( R_k \) be the reaction of which we want to quantify the dependence of \( S(c) \) on its reaction rate \( c_k \). The goal of sensitivity analysis is to compute the partial derivative (called sensitivity coefficient) of \( S(c) \) with respect to the reaction rate \( c_k \), i.e., \( \partial S(c)/\partial c_k \). The sensitivity coefficient \( \partial S(c)/\partial c_k \) can be estimated by applying a small scalar perturbation \( \epsilon_k \) to the reaction rate \( c_k \). Let \( \epsilon_k \) be a unit \( m \)-vector in which the \( k \)th element is 1, while other elements are 0s. The sensitivity coefficient with respect to a reaction rate \( c_k \) can be approximated by the forward difference

\[
\frac{\partial S(c)}{\partial c_k} \approx \frac{S(c + \epsilon_k e_k) - S(c)}{\epsilon_k} \\
\approx \frac{\mathbb{E}[f(X^{c+\epsilon_k e_k}(t))] - \mathbb{E}[f(X^c(t))]}{\epsilon_k}
\]

The bias of the forward difference due to the truncation error is \( O(\epsilon_k^2) \). We note that it can reduce the bias to \( O(\epsilon_k^2) \) by using the centered finite difference \([53]\). In Eq. 7 the bias becomes zero in the limit that \( \epsilon_k \to 0 \).

The estimator for the forward difference in Eq. 7 can be constructed as

\[
Z = \frac{1}{N} \sum_{i=1}^{N} \frac{f(X_{[i]}^{c+\epsilon_k e_k}(t)) - f(X_{[i]}^{c}(t))}{\epsilon_k}
\]

where \( N \) is the number of simulation runs and \( X_{[i]}^{c}(t) \) denotes the \( i \)th realization with rate parameter \( c \). A naive implementation of the estimator where \( X_{[i]}^{c}(t) \) and \( X_{[i]}^{c+\epsilon_k e_k}(t) \) are generated independently will produce a large variance. In fact, the
variance of the estimator \( \text{var}[Z] \) in the naive implementation is equal to the sum of two variances \( \text{var}[f(X^{c+\epsilon_k}e_k(t))] \) and \( \text{var}[f(X^{c}(t))] \) because their covariance is zero (i.e., \( \text{cov}[f(X^{c+\epsilon_k}e_k(t)), f(X^{c}(t))] = 0 \)). CRN and CFD reduce the variance of the estimator by introducing a (positive) correlation between the \( X^{c}(t) \) and \( X^{c+\epsilon_k}e_k(t) \), hence \( \text{cov}[f(X^{c+\epsilon_k}e_k(t)), f(X^{c}(t))] \neq 0 \), during the simulation of these processes.

The CRN method correlates \( X^{c}(t) \) and \( X^{c+\epsilon_k}e_k(t) \) by using the same stream of random numbers during the simulation. Algorithm 1 outlines the steps of CRN approach as applied to SSA. The key of the CRN is that the random generator used in both the simulations of \( X^{c}(t) \) and \( X^{c+\epsilon_k}e_k(t) \) is initialized with the same seed \( w \).

\[ \begin{align*}
    \text{Algorithm 1 Common Random Number method (CRN)} \\
    1: & \quad \text{define reaction index } k \text{ for sensitivity analysis} \\
    2: & \quad \text{initialize time } t = 0 \text{ and state } X^{c} = X^{c+\epsilon_k}e_k = x_0 \\
    3: & \quad \text{generate a random seed } w \\
    4: & \quad \text{seed the random number generator with } w \\
    5: & \quad \text{realize } X^{c} \text{ by performing SSA until time } T_{\text{max}} \\
    6: & \quad \text{reseed the random number generator with } w \\
    7: & \quad \text{realize } X^{c+\epsilon_k}e_k \text{ by performing SSA until time } T_{\text{max}}
\end{align*} \]

The CFD employs the random time change (RTC) representation to correlate \( X^{c}(t) \) and \( X^{c+\epsilon_k}e_k(t) \). Specifically, by RTC representation and additive property of the Poisson process, it gives

\[
X^{c}(t) = X^{c}(0) + \sum_{j=1}^{m} Y_{j,1}\int_{0}^{t} b_{j}(s)ds v_{j} + \sum_{j=1}^{m} Y_{j,2}\int_{0}^{t} (a_{j}(X^{c}(s)) - b_{j}(s))ds v_{j}
\]

(9)

and

\[
X^{c+\epsilon_k}e_k(t) = X^{c+\epsilon_k}e_k(0) + \sum_{j=1}^{m} Y_{j,1}\int_{0}^{t} b_{j}(s)ds v_{j} + \sum_{j=1}^{m} Y_{j,3}\int_{0}^{t} (a_{j}(X^{c+\epsilon_k}e_k(s)) - b_{j}(s))ds v_{j}
\]

(10)

where \( b_{j}(t) = \min(a_{j}(X^{c}(t)), a_{j}(X^{c+\epsilon_k}e_k(t))) \) and \( Y_{j,i} \) for \( j = 1, \ldots, m \) and \( i \in \{1, 2, 3\} \) denote independent unit-rate Poisson processes. The representation of the nominal and perturbed processes in Eqs. 9-10 is the key of the CFD method outlined in Algorithm 2 where the simulation of these processes are shared by the Poisson process \( Y_{j,1}\int_{0}^{t} b_{j}(s)ds \). We note that a similar method to CFD has been also proposed in Rathinam et al. [44].
Algorithm 2 Coupled Finite Difference method (CFD)

1: define reaction index $k$ for sensitivity analysis
2: initialize time $t = 0$ and state $X^c = X^{c+\varepsilon_k e_k} = x_0$
3: set $T_{j,i} = 0$ and $P_{j,i} = \ln(1/r_{j,i})$ where $r_{j,i} \sim U(0,1)$ for $j = 1, \ldots, m$ and $i \in \{1,2,3\}$
4: while ($t < T_{\text{max}}$) do
5: compute $a_j(X^c + \varepsilon_k e_k)$ and $a_j(X^c(t))$ for $j = 1, \ldots, m$
6: set $b_{j,1} = \min(a_j(X^c + \varepsilon_k e_k), a_j(X^c(t)))$ and compute $b_{j,2} = a_j(X^c) - b_{j,1}$ and $b_{j,3} = a_j(X^c + \varepsilon_k e_k) - b_{j,1}$ for $j = 1, \ldots, m$
7: compute $\tau_{j,i} = (P_{j,i} - T_{j,i})/b_{j,i}$ for $j = 1, \ldots, m$ and $i = 1, \ldots, 3$
8: set $\tau = \min(\tau_{j,i})$ and let $(\mu, \alpha)$ be the pair of indices where the minimum is selected
9: set $t = t + \tau$
10: if ($\alpha == 1$) then
11: update $(X^c, X^{c+\varepsilon_k e_k}) = (X^c, X^{c+\varepsilon_k e_k}) + (v_\mu, v_\mu)$
12: else if ($\alpha == 2$) then
13: update $X^c = X^c + v_\mu$
14: else if ($\alpha == 3$) then
15: update $X^{c+\varepsilon_k e_k} = X^{c+\varepsilon_k e_k} + v_\mu$
16: end if
17: set $T_{j,i} = T_{j,i} + b_{j,i}\tau$ for $j = 1, \ldots, m$ and $i \in \{1,2,3\}$
18: set $P_{\mu,\alpha} = P_{\mu,\alpha} + \ln(1/r)$ where $r \sim U(0,1)$
19: end while
3. Sensitivity analysis using rejection-based approach

This section introduces a new finite difference method for efficiently computing sensitivities of biochemical reactions. We first present the theoretical background of the rejection-based stochastic simulation algorithm in which the selection of the next reaction firing is based on propensity bounds of reactions and the acceptance-rejection technique. Then, we describe our new rejection-based finite difference (RFD) method that employs the concept of propensity bounds of reactions to correlate the simulation of the nominal and perturbed processes.

3.1. Background on rejection-based simulation

The rejection-based SSA (RSSA) is an exact stochastic simulation algorithm that aims to reduce the average number of propensity calculations, hence improving simulation performance (see Thanh et al. [25] for a formal proof of the exactness of RSSA). The mathematical framework for the selection of reaction firings in RSSA is a rejection-based sampling technique. It uses propensity bounds \([a_j, a_j]\), which delimits all possible values of the propensity \(a_j(X(t))\) of each reaction \(R_j\) with \(j = 1, \ldots, m\), to select the next reaction firing. The propensity bounds are computed by bounding the state \(X(t)\) to the fluctuation interval \([\underline{X}, \overline{X}]\) such that the inequality \(\underline{X} \leq X(t) \leq \overline{X}\) holds for each species \(S_i\), with \(i = 1, \ldots, n\), in the state \(X(t)\).

RSSA selects the next reaction firing using propensity bounds in two steps. First, a candidate reaction \(R_{\mu}\) is selected with probability \(\frac{a_{\mu}}{a_0}\) where \(a_0 = \sum_{j=1}^{m} a_j\). Second, the candidate \(R_{\mu}\) is validated through a rejection test with success probability \(\frac{a_{\mu}(X(t))}{a_{\mu}}\). If the candidate \(R_{\mu}\) passes the rejection test, then it is accepted to fire. Otherwise, it is rejected and another candidate is selected to test. The rejection test requires to compute propensity \(a_{\mu}(X(t))\), but the implementation can postpone the computation by exploiting the fact that if a candidate reaction \(R_{\mu}\) is accepted with probability \(\frac{a_{\mu}}{a_{\mu}}\), then it can be accepted without evaluating \(a_{\mu}(X(t))\) because of \(\frac{a_{\mu}}{a_{\mu}} < \frac{a_{\mu}(X(t))}{a_{\mu}}\).

Let \(k\) be the number of trials such that the first \(k-1\) trials are rejections and \(R_{\mu}\) is accepted at the \(k\)th trial. The firing time \(\tau\) of the accepted candidate \(R_{\mu}\) in RSSA is the sum of \(k\) independent exponentially distributed numbers with the same rate \(\overline{a}_0\), which is equivalent to an \(\text{Erlang}(k, \overline{a}_0)\) distribution. RSSA thus generates the firing time \(\tau\) of the accepted candidate \(R_{\mu}\) by sampling the corresponding \(\text{Erlang}\) distribution.

3.2. Rejection-based finite difference method

This section introduces the rejection-based finite difference (RFD) method for performing sensitivity analysis of biochemical reactions. Our method uses propensity bounds of reactions to correlate the nominal and perturbed processes and employs the rejection-based mechanism to correct the selection. By using such propensity bounds, RFD is able to perturb many reaction rates simultaneously.

Let \(Q\) be the set of reaction indices for which sensitivities are to be computed. Let \(X^c(t)\) be the state of the nominal process and \(X^{c+\epsilon_k e_k}(t)\), for \(k \in Q\), be state of perturbed process. Note that we denote the change in \(k\)th element of the rate vector \(c\) with \(\epsilon_k e_k\). For each reaction \(R_j\) with \(j = 1, \ldots, m\), let \([a_j, a_j]\) be a interval that bounds all...
values of the propensity function $a_j$ of reaction $R_j$ over all the nominal and perturbed states. i.e., $a_j \leq a_j(X^c(t))$, $\{a_j(X^{c+\epsilon \epsilon_k}(t))\}_{k \in \mathcal{Q}} \leq \overline{a}_j$. Because all reactions in the nominal and perturbed processes share the same propensity upper bound $\overline{a}_j$ and lower bound $a_j$ with $j = 1, \ldots, m$, these bounds can be used to select the candidate reaction. During the rejection-based step, the candidate reaction can be accepted or rejected depending on the value of the propensity of the reaction in the corresponding process. We note that the original rejection-based selection in RSSA is composed of many steps in which only the candidate which is accepted is assigned the firing time. However, in computing sensitivities we need to synchronize the nominal and perturbed processes after each trial. To cope with this, the key of RFD is to decompose the rejection-based selection into single trials and assign time stamp for each trial. The derivation of propensity bounds and selection of reaction firings in simulating the nominal and perturbed processes by RFD are as follows.

The computation of the propensity bound $[a_j, \overline{a}_j]$ that bounds all values of the propensity function $a_j$ of reaction $R_j$ with $j = 1, \ldots, m$ over the nominal state $X^c(t)$ and the perturbed states $X^{c+\epsilon \epsilon_k}(t)$ with $k \in \mathcal{Q}$ is done by bounding the rate $c_j$ of reaction $R_j$ as well as constraining all populations of its reactant species in $X^c(t)$ and $X^{c+\epsilon \epsilon_k}(t)$ with $k \in \mathcal{Q}$ into the fluctuation interval $[X, \overline{X}]$.

For each reaction $R_j$, RFD defines a lower value $c_j$ and an upper value $\overline{c}_j$ as the minimum and maximum of $c_j$ and $c_j + \epsilon_j$ to bound the reaction rate. Specifically, it sets $c_j = \min(c_j, c_j + \epsilon_j)$ and $\overline{c}_j = \max(c_j, c_j + \epsilon_j)$ with $j = 1, \ldots, m$. We note that if $j \notin \mathcal{Q}$ then $c_j = \overline{c}_j = c_j$.

The derivation of the lower bound $X_i^c$ and the upper bound $\overline{X}_i$ of populations of each species $S_i$, $i = 1, \ldots, n$, in the nominal state $X^c(t)$ and perturbed states $X^{c+\epsilon \epsilon_k}(t)$ for $k \in \mathcal{Q}$, hence forming the fluctuation interval $[X, \overline{X}]$, is obtained by constraining the minimum and maximum population of this species in all these states. Precisely, RFD defines $X_i^{c, m in} = \min(X_i^c(t), \{X_i^{c+\epsilon \epsilon_k}(t)\}_{k \in \mathcal{Q}})$ and $X_i^{c, m ax} = \max(X_i^c(t), \{X_i^{c+\epsilon \epsilon_k}(t)\}_{k \in \mathcal{Q}})$ as the minimum and maximum population of species $S_i$, respectively. The computation of the population bounds for $S_i$ is thus $X_i = (1 - \delta_i)X_i^{m in}$ and $\overline{X}_i = (1 + \delta_i)X_i^{m ax}$ where the fluctuation rate $\delta_i$ is a parameter. Note that the fluctuation rate $\delta_i$ can be chosen arbitrarily without affecting the simulation result, but only the simulation performance. For typical models, the fluctuation rate chosen around 10% to 20% gives better performance (see numerical examples in Section 5).

The propensity lower bound $a_j$ and upper bound $\overline{a}_j$ for each reaction $R_j$, with $j = 1, \ldots, m$, is computed by optimizing the propensity function $a_j$ over the rate bound $[c_j, \overline{c}_j]$ and fluctuation interval $[X, \overline{X}]$. For mass-action propensity function $a_j$ given in Eq. 2, the bounds can be computed easily using its monotonic property and interval arithmetic [54]. Specifically, let $h_j$ and $\overline{h}_j$ be the minimum and maximum of function $h_j$ over the fluctuation interval $[X, \overline{X}]$, respectively. By monotonic property of function $h_j$, it gives $h_j = h_j(X)$ and $\overline{h}_j = h_j(\overline{X})$. Then by interval analysis, the propensity bounds of $R_j$ can be computed as

$$a_j = c_j h_j$$

(11)
and
\[ \alpha_j = c_j r_j \]  

Having the propensity bounds \([\alpha_j, \bar{\alpha}_j]\) of all reactions, RFD selects the reaction firings to update the states \(X^c\) or \(X^{c+e_\ell v_k}\) for \(k \in Q\) by decomposing the complex rejection-based selection into single trials. Specifically, in each trial a candidate reaction \(R_\mu\) is selected with a discrete probability \(\frac{\alpha_j}{\bar{\alpha}_0}\) and its time stamp \(\tau\) is generated following an exponential distribution \(\text{Exp}(\bar{\alpha}_0)\) where \(\bar{\alpha}_0 = \sum_{j=1}^m \bar{\alpha}_j\). For the time \(\tau\), it can be calculated by the inverse transformation as \(\tau = (1/\bar{\alpha}_0) \ln(1/r_1)\) where \(r_1 \sim U(0, 1)\). The selection of candidate reaction \(R_\mu\) with discrete probability \(\frac{\alpha_j}{\bar{\alpha}_0}\) can be performed by linearly accumulating propensity upper bounds until it finds the smallest reaction index \(\mu\) satisfying the inequality: \(\sum_{j=1}^\mu \bar{\alpha}_j > r_2 \cdot \bar{\alpha}_0\) where \(r_2 \sim U(0, 1)\). For large networks, more efficient search algorithms can be applied to improve the performance [26]. Knowing candidate reaction \(R_\mu\), RFD applies the rejection-based test to decide whether to update the states in the trial. This rejection-based test ensures that marginal distributions of the nominal and perturbed processes are correct, although their joint distribution is correlated. For this purpose, a random number \(r_3 \sim U(0, 1)\) is generated. If \(r_3 \leq \frac{\alpha_j}{\bar{\alpha}_\mu}\) holds true, then both the states \(X^c(t)\) and \(X^{c+e_\ell v_k}\), for \(k \in Q\), are updated. If the test fails, RFD computes the propensities of \(R_\mu\) corresponding to the each state and performs the check again with \(r_3\). More in details, RFD computes the propensity \(a_\mu(X^c(t))\) as well as \(a_\mu(X^{c+e_\ell v_k}(t))\) for each \(k \in Q\). Then, it checks whether \(r_3 \leq a_\mu(X^c(t))/\bar{\alpha}_\mu\) (respectively, \(r_3 \leq a_\mu(X^{c+e_\ell v_k}(t))/\bar{\alpha}_\mu\)) in order to update \(X^c(t)\) (respectively, \(X^{c+e_\ell v_k}(t)\)).

Algorithm 3 outlines the simulation of RFD for generating the realizations of \(X^c\) and \(X^{c+e_\ell v_k}\) with \(k \in Q\). The computation of the propensity bounds \(\alpha_j\) and \(\bar{\alpha}_\mu\) by bounding the reaction rates and population of species is performed in line \(3\). To facilitate update of propensity bounds when a species whose population moves out of the fluctuation interval, the algorithm makes use of the Species-Reaction (SR) graph [25] to retrieve which reactions should update their propensity bounds when a species exits its fluctuation interval. The SR dependency graph \(G\) is a directed bipartite graph which shows the dependency of reactions on species. A directed edge from a species \(S_i\) to a reaction \(R_j\) is in the graph if a change in the population of species \(S_i\) requires reaction \(R_j\) to recompute its propensity. The SR dependency graph \(G\) is built in line \(2\).

The main simulation loop of the RFD algorithm is composed of two main tasks. The first task selects the reaction to update the states using propensity bounds and rejection-based mechanism. The second task updates propensity bounds of reactions when there exists a species whose population moves out of the current fluctuation interval. The loop is repeated until the time \(t\) passes a predefined time \(T_{max}\).

The selection of the reaction to update the states is implemented in lines \([10 - 20]\). For each loop, three random numbers \(r_1, r_2\) and \(r_3 \sim U(0, 1)\) are generated in which \(r_1\) is used to compute \(\tau\), while \(r_2\) and \(r_3\) is used to select the candidate and to validate the candidate. The selection is looped until there is a species whose population moves out of the fluctuation interval due to reaction firings. In this case, a new fluctuation interval as well as propensity bounds of reactions should be updated.

The update of propensity bounds is outlined in lines \([30 - 40]\). The species that should update their fluctuation interval is kept track by the set \(\text{UpdateSpeciesSet}\), which
Algorithm 3 Rejection-based Finite difference method (RFD)

1: define set $Q$ containing reaction indices for sensitivity analysis
2: build the species-reaction (SR) dependency graph $\mathcal{G}$
3: initialize time $t = 0$ and state $X^c = X^{c+\epsilon ek} = x_0$ for all $k \in Q$
4: compute rate bounds $c_j$ and $\overline{c_j}$ for reaction $R_j$ with $j = 1, \ldots, m$
5: compute fluctuation interval $[X_i, \overline{X}_i]$ that bounds the population of species $S_i$, $i = 1 \ldots n$, in all states.
6: compute propensity bounds $a_j$ and $\overline{a_j}$ for reaction $R_j$ with $j = 1, \ldots, m$
7: set $a_0 = \sum_{j=1}^{m} a_j$
8: while $(t < T_{\text{max}})$ do
9:   set $\text{UpdateSpeciesSet} = \emptyset$
10:   while (populations of each species $S_i$ in states are in $[X_i, \overline{X}_i]$) do
11:      generate three random numbers $r_1, r_2$ and $r_3 \sim U(0, 1)$
12:      set $\tau = (1/r_1) \ln(1/a_0)$
13:      update time $t = t + \tau$
14:      select minimum index $\mu$ s.t. $\sum_{j=1}^{\mu} \overline{a_j} > r_2 a_0$
15:      if $(r_3 \leq a_{\mu}/\overline{a_{\mu}})$ then
16:         update $X^c$ and $X^{c+\epsilon ek}$ for all $k \in Q$ by $v_{\mu}$
17:      else
18:         compute $a_{\mu}(X^c)$
19:         if $(r_3 \leq a_{\mu}(X^c)/\overline{a_{\mu}})$ then
20:            update $X^c = X^c + v_{\mu}$
21:         end if
22:      end if
23:      for all $(k \in Q)$ do
24:         compute $a_{\mu}(X^{c+\epsilon ek})$
25:         if $(r_3 \leq a_{\mu}(X^{c+\epsilon ek})/\overline{a_{\mu}})$ then
26:            update $X^{c+\epsilon ek} = X^{c+\epsilon ek} + v_{\mu}$
27:         end if
28:      end for
29:   end while
30:   for all (species $S_i$ where population $X_i^c$ or $X_i^{c+\epsilon ek}$, with $k \in Q, \notin [X_i, \overline{X}_i]$) do
31:      set $\text{UpdateSpeciesSet} = \text{UpdateSpeciesSet} \cup \{S_i\}$
32:   end for
33:   for all (species $S_i \in \text{UpdateSpeciesSet}$) do
34:      define a new fluctuation interval $[X_i, \overline{X}_i]$
35:      extract reactions $\text{ReactionsAffectedBy}(S_i)$ affected by $S_i$ from SR graph $\mathcal{G}$
36:      for all ($R_j \in \text{ReactionsAffectedBy}(S_i)$) do
37:         compute new propensity bounds $\overline{a_j}$ and $a_j$
38:      end for
39:   end for
40: end while
41: end if

11
is initialized to be an empty set at the beginning in line 9. For each species $S_i \in \text{UpdateSpeciesSet}$, a new fluctuation interval $[X_i^-, X_i^+]$ that bounds all populations of the species in all states is computed. Then, reactions affected by $S_i$, which is denoted by the set $\text{ReactionsAffectedBy}(S_i)$, is extracted from SR graph $G$. For each $R_j \in \text{ReactionsAffectedBy}(S_i)$, its new lower bound $a_j$ and upper bound $\overline{a_j}$ is recomputed.

4. Numerical examples

We report in this section the numerical results by our RFD algorithm in comparison with CRN and CFD algorithms. For CRN and CFD, we use the dependency graph [12] to decide which reactions should update their propensities when a reaction fires. All algorithms in this section are implemented in Java and run on a Intel i5-540M processor. The implementation of algorithms as well as the benchmark models are freely available at the url http://www.cosbi.eu/research/prototypes/rssa. We compare these methods in two models that are: the birth-death process and the Rho GTP-binding protein model. The former is a simple model, where the exact form of the sensitivity analysis is available, while the latter case is a large model where simulation must be used to perform sensitivity analysis. These models are used to demonstrate the advantages of RFD to cope with different aspects of biochemical reactions.

4.1. Birth death process

The birth-death process is a simple model, but is commonly found in applications. The model has two reactions that describe the producing and consuming of a species $S$. The reactions of the model are listed in Eq. 13.

$$\emptyset \xrightarrow{c_1} S \xrightarrow{c_2} \emptyset$$

(13)

The species $S$ is created with rate $c_1$ and degraded with rate $c_2$. The propensities of reactions in birth-death process is assumed to follow mass-action kinetics, hence $a_1 = c_1$ and $a_2 = c_2\#S$. Let $s_0$ be the initial population of $S$ at time $t = 0$. For this model, the population of species $S$ at time $t$ can be computed analytically [55] as the sum of Binomial distribution $\text{Bin}(s_0, p)$ and Poisson distribution $\text{Poi}(\lambda)$ where $p = e^{-c_2 t}$ and $\lambda = (c_1/c_2)(1 - e^{-c_2 t})$. The expected value of number of species $S$ at a time $t$ is thus given by

$$E[\#S(t)] = s_0 e^{-c_2 t} + \left(\frac{c_1}{c_2}\right)(1 - e^{-c_2 t})$$

(14)

For the computation of the sensitivities of the population of species $S$ by CRN, CFD and RFD, the nominal rates of reactions are set to $c_1 = 100$ and $c_2 = 1$. The initial population of $S$ is set $s_0 = 100$ and the simulation time is $T_{\text{max}} = 100$. For RFD, the fluctuation interval of species $S$ is defined to be $\pm 10\%$ of its population.

First, we compute sensitivities of the population of species $S$ by increasing the reaction rate $c_2$ an amount of $\epsilon = 10\%$. Figure [1] depicts the estimated sensitivity of the population of species $S$ by CRN, CFD and RFD by $N = 1000$ simulation runs. Figure [2] gives the standard deviation for the estimators with varying the number of simulation runs. The figures show that accuracy of CFD and RFD are better than CRN.
Figure 1: Sensitivities of the expected value of population of species S by CRN, CFD and RFD methods in comparison with exact value. The nominal reaction rate $c_2$ is increased by $\epsilon_2 = 10\%$. The sensitivity values are obtained by performing 1000 runs of these methods each with simulation time $T_{\text{max}} = 100$.

Figure 2: Standard deviations in estimating sensitivities of the expected value of population of species S by perturbing the reaction rate $c_2$ by an amount $\epsilon_2 = 10\%$ using CRN, CFD and RFD methods with different number of simulation runs.

Figure 3 shows performance of CRN, CFD and RFD in computing the sensitivities of the population of species S by increasing the reaction rate $c_2$ an amount of $\epsilon_2 = 10\%$. RFD has a similar performance as CFD, while it is about 2 times faster than CRN. The performance gain by RFD in comparison with CRN is obtained by reduces the number of simulation steps and the number of propensity updates. The number of simulation steps, hence the number of propensity updates, performed by CRN is $4.0 \times 10^4$ and by CFD is $2.06 \times 10^4$. RFD performs $2.21 \times 10^4$ simulation steps, but only has to update propensity bounds 160 times.
Figure 3: Performance of CRN, CFD and RFD methods in computing sensitivities of species $S$ where the nominal reaction rate $c_2$ is increased by $\epsilon_2 = 10\%$.

To demonstrate the computational efficiency of RFD, we repeat the sensitivity analysis of the population of species $S$ by simultaneously perturbing both the reaction rate $c_1$ by $\epsilon_1 = 1\%$ and the reaction rate $c_2$ by $\epsilon_2 = 10\%$. The performance is averaged by $N = 1000$ simulation runs. In this experiment, because there are 4 combinations of perturbation parameters (one of such combinations is shown in the previous experiment in Figs. [1]-[3], CRN and CFD have to repeat the computation 4 times corresponding to each combination. In contrast, RFD is able to perturb two reaction rates simultaneously. Figure 4 shows performance of CRN, CFD and RFD by simultaneously perturbing both the reaction rate $c_1$ and the reaction rate $c_2$. The figure shows that RFD is significantly more efficient than CRN and CFD. Specifically, the computational time of CRN and CFD is thus nearly 4 times increased in comparison with the case where only one parameter is perturbed in the previous experiment. RFD in this setting performs $2.25 \times 10^4$ simulation steps which are similar to the case where only one parameter is perturbed. The increased computational time of RFD in comparison with the case where only one parameter is perturbed is due to more states keeping track. The result is the performance of RFD is about 3.89 times and 2 times faster than CRN and CFD, respectively.

4.2. Rho GTP-binding protein model

We use the model of Rho GTP-binding proteins [56, 57] to demonstrate the computational efficiency of RFD in applying to large models. The Rho GTP-binding proteins constitute a subgroup of the Ras super-family of GTP hydrolases (GTPases) that regulate the transmission of external stimuli to effectors. The Rho GTP-binding protein cycle switches between inactive and active states depending upon binding of either GDP or GTP to the GTPases, respectively. The cycle is controlled by two regulatory proteins: guanine nucleotide exchange factors (GEFs) and GTPase-activating proteins (GAPs). GEFs promote the GDP dissociation and GTP binding, hence producing the activation of the GTPase. In contrast, GAPs stimulate the hydrolysis of the bound GTP.
molecules, hence transferring the GTPase back to the inactive state. In the active state, Rho GTP-binding proteins interact and activate downstream effectors.

Table 1 lists the reactions and the rates of the Rho GTP-binding protein model. In the model, R denotes the Rho GTP-binding protein in nucleotide free form and RD and RT denote its GDP and GTP bound forms, respectively. A and E denote GAP and GEF, respectively. The model has 23 reactions. The initial populations for species are \( \#R = 1000 \), \( \#E = 776 \) and \( \#A = 10 \), while it is zero for all other species.

Figure 5 shows the sensitivities of the species RE and RD by increasing the rate \( c_1 \) by an amount \( \epsilon_1 = 20\% \). The result is obtained by performing 1000 runs of CRN, CFD and RFD with simulation time \( T_{\text{max}} = 10 \).

Figure 6 depicts the performance of CRN, CFD and RFD. For this experiment, the number of simulation steps performed by CRN, CFD and RFD is \( 2.88 \times 10^5 \), \( 1.45 \times 10^5 \) and \( 2.53 \times 10^5 \), respectively. The reason for the low performance of CFD in this experiment, even though it performs only a half number of simulation steps in comparison with CRN, is due to the high cost for the update of propensities and related data structures after reaction firings. By reducing the propensity updates during the simulation, RFD significantly improves the performance. Specifically RFD only performs \( 2.13 \times 10^4 \) propensity updates (about 9% of its simulation steps). The result is the performance of RFD is 2.2 and 2.7 times faster than CRN and CFD, respectively.

5. Conclusions

This paper proposed a new rejection-based finite difference (RFD) method for estimating sensitivities of biochemical reactions. Our method uses propensity bounds of reactions and rejection-based mechanism to construct the estimator. The propensity bounds of a reaction are derived by bounding the population of reactant species and
Table 1: Rho GTP-binding model

| Reaction | Rate |
|----------|------|
| $R_1$: A + R → RA | $c_1 = 1$ |
| $R_2$: A + RD → RDA | $c_2 = 1$ |
| $R_3$: A + RT → RTA | $c_3 = 1$ |
| $R_4$: E + R → RE | $c_4 = 0.43$ |
| $R_5$: E + RD → RDE | $c_5 = 0.0054$ |
| $R_6$: E + RT → RTE | $c_6 = 0.0075$ |
| $R_7$: R → RD | $c_7 = 1.65$ |
| $R_8$: R → RT | $c_8 = 50$ |
| $R_9$: RA → A + R | $c_9 = 500$ |
| $R_{10}$: RD → R | $c_{10} = 0.02$ |
| $R_{11}$: RDA → A + RD | $c_{11} = 500$ |
| $R_{12}$: RDE → E + RD | $c_{12} = 0.136$ |
| $R_{13}$: RDE → RE | $c_{13} = 6.0$ |
| $R_{14}$: RE → E + R | $c_{14} = 1.074$ |
| $R_{15}$: RE → RDE | $c_{15} = 1.65$ |
| $R_{16}$: RE → RTE | $c_{16} = 50$ |
| $R_{17}$: RT → R | $c_{17} = 0.02$ |
| $R_{18}$: RT → RD | $c_{18} = 0.02$ |
| $R_{19}$: RTA → A + RT | $c_{19} = 3$ |
| $R_{20}$: RTA → RDA | $c_{20} = 2104$ |
| $R_{21}$: RTE → E + RT | $c_{21} = 76.8$ |
| $R_{22}$: RTE → RDE | $c_{22} = 0.02$ |
| $R_{23}$: RTE → RE | $c_{23} = 0.02$ |
Figure 5: Sensitivities of the expected value of population of species RA and RD by CRN, CFD and RFD methods in comparison with exact value. The nominal reaction rate $c_1$ is increased by $\epsilon_1 = 20\%$. The sensitivity values are obtained by performing 1000 runs of these methods each with simulation time $T_{\text{max}} = 10$.

Figure 6: Performance of CRN, CFD and RFD methods in computing sensitivities of the expected value of population of species RA and RD where the reaction rate $c_1$ is increased by $\epsilon_1 = 20\%$.
its rate. By employing propensity bounds of reactions, the simulations of the reactions with nominal and perturbed rates can be correlated, hence reducing of the variance of the estimator. By employing propensity bounds of reactions to couple nominal and perturbed processes, our method allows to simultaneously perturbing many reaction rates at a time. The exactness of the simulation is recovered by applying the rejection-based mechanism. The computational gain of our method is achieved by reducing the propensity updates during the simulation.

References

[1] Harley H. McAdams and Adam Arkin. It’s a noisy business! genetic regulation at the nanomolar scale. Trends in Genetics, 15(2), 1999.

[2] Harley H. McAdams and Adam Arkin. Stochastic mechanisms in gene expression. In PNAS, 94(3):814–819, 1997.

[3] J. Vilar, H. Kueh, N. Barkai and S. Leibler. Mechanisms of noise-resistance in genetic oscillators. In PNAS, 99(9):5988–5992, 2002.

[4] Ertugrul M. Ozbudak, Mukund Thattai, Iren Kurtser, Alan D. Grossman and Alexander van Oudenaarden. Regulation of noise in the expression of a single gene. Nature Genetics, 31:69–73, 2002.

[5] Michael B. Elowitz and Arnold J. Levine and Eric D. Siggia and Peter S. Swain. Stochastic Gene Expression in a Single Cell. Science, 297:1183, 2002.

[6] Adam Arkin, John Ross and Harley H. McAdams. Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected escherichia coli cells. Genetics, 149(4):1633–1648, 1998.

[7] Juan M. Pedraza and Alexander van Oudenaarden. Noise propagation in gene networks. Science, 307(5717):1965–1969, 2005.

[8] Jonathan M. Raser and Erin K. O’Shea. Noise in gene expression: Origins, consequences and control. Science, 309:2010–2013, 2005.

[9] Daniel T. Gillespie. A rigorous derivation of the chemical master equation. Physica A, 188(1-3):404–425, 2007.

[10] Daniel T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. J. Comp. Phys., 22(4):403–434, 1976.

[11] Daniel T. Gillespie. Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem., 81(25):2340–2361, 1977.

[12] Michael Gibson and Jehoshua Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. J. Phys. Chem. A, 104(9):1876–1889, 2000.
[13] David F. Anderson. A modified next reaction method for simulating chemical systems with time dependent propensities and delays. *J. Chem. Phys.*, 127(21):214107, 2007.

[14] Yang Cao, Hong Li, and Linda Petzold. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *J. Chem. Phys.*, 121(9):4059, 2004.

[15] James McCollum and et al. The sorting direct method for stochastic simulation of biochemical systems with varying reaction execution behavior. *Comp. Bio. Chem.*, 30(1):39–49, 2006.

[16] James Blue, Isabel Beichl, and Francis Sullivan. Faster monte carlo simulations. *Phys. Rev. E*, 51(2):867–868, 1995.

[17] Hong Li and Linda Petzold. Logarithmic direct method for discrete stochastic simulation of chemically reacting systems. Technical Report [https://cse.cs.ucsb.edu/sites/cse.cs.ucsb.edu/files/publications/ldm0513.pdf](https://cse.cs.ucsb.edu/sites/cse.cs.ucsb.edu/files/publications/ldm0513.pdf), 2006.

[18] Vo H. Thanh and Roberto Zunino. Tree-based search for stochastic simulation algorithm. In *Proc. of ACM-SAC*, pages 1415–1416, 2012.

[19] Vo H. Thanh and R. Zunino. Adaptive tree-based search for stochastic simulation algorithm. *Intl. J. Comp. Bio. and Drug Des.*, 7(4):341–357, 2014.

[20] S. Mauch and M. Stalzer. Efficient formulations for exact stochastic simulation of chemical systems. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 8(1):27–35, 2011.

[21] Tim Schulze. Efficient kinetic monte carlo simulation. *J. Comp. Phys.*, 227(4):2455–2462, 2008.

[22] Alexander Slepoy, Aidan P. Thompson, and Steven J. Plimpton. A constant-time kinetic monte carlo algorithm for simulation of large biochemical reaction networks. *J. Chem. Phys.*, 128(20):205101, 2008.

[23] Rajesh Ramaswamy, Nlido Gonzalez-Segredo, and Ivo F. Sbalzarini. A new class of highly efficient exact stochastic simulation algorithms for chemical reaction networks. *J. Chem. Phys.*, 130(24):244104, 2009.

[24] Sagar Indurkhya and Jacob Beal. Reaction factoring and bipartite update graphs accelerate the gillespie algorithm for large-scale biochemical systems. *PLoS ONE*, 5(1):8125, 2010.

[25] Vo H. Thanh, Corrado Priami, and Roberto Zunino. Efficient rejection-based simulation of biochemical reactions with stochastic noise and delays. *J. Chem. Phys.*, 141(13), 2014.

[26] Vo H. Thanh, Roberto Zunino, and Corrado Priami. On the rejection-based algorithm for simulation and analysis of large-scale reaction networks. *J. Chem. Phys.*, 142(24):244106, 2015.
[27] Vo H. Thanh, Roberto Zunino, and Corrado Priami. Efficient constant-time complexity algorithm for stochastic simulation of large reaction networks. *IEEE/ACM Trans. Comput. Biol. Bioinf.*, 2016 (in press).

[28] Vo H. Thanh. *On Efficient Algorithms for Stochastic Simulation of Biochemical Reaction Systems*. PhD thesis, University of Trento, Italy. [http://eprints-phd.biblio.unitn.it/1070/](http://eprints-phd.biblio.unitn.it/1070/), 2013.

[29] Werner Sandmann. Discrete-time stochastic modeling and simulation of biochemical networks. *Comput. Biol. Chem.*, 32(4):292, 2008.

[30] Hong Li and Linda Petzold. Efficient parallelization of the stochastic simulation algorithm for chemically reacting systems on the graphics processing unit. *Intl. J. of High Performance Computing Applications*, 24(2):107–116, 2010.

[31] Vo H. Thanh and Roberto Zunino. Parallel stochastic simulation of biochemical reaction systems on multi-core processors. In *Proc. of CSSim*, pages 162–170, 2011.

[32] Dmitri Bratsun, Dmitri Volfson, Lev S. Tsimring, and Jeff Hasty. Delay-induced stochastic oscillations in gene regulation. *PNAS*, 102:14593–14598, 2005.

[33] Manuel Barrio, Kevin Burrage, André Leier, and Tianhai Tian. Oscillatory regulation of hes1: discrete stochastic delay modelling and simulation. *PLoS Comput. Biol.*, 2(9):e117, 2006.

[34] Xiaodong Cai. Exact stochastic simulation of coupled chemical reactions with delays. *J. Chem. Phys.*, 126(12):124108, 2007.

[35] Vo H. Thanh, Roberto Zunino and Corrado Priami. Efficient Stochastic Simulation of Biochemical Reactions with Noise and Delays. *J. Chem. Phys.*, 146(8):084107, 2017.

[36] Ting Lu, Dmitri Volfson, Lev Tsimring, and Jeff Hasty. Cellular growth and division in the gillespie algorithm. *IEE Systems Biology*, 1(1):121–128, 2004.

[37] Vo H. Thanh and Corrado Priami. Simulation of biochemical reactions with time-dependent rates by the rejection-based algorithm. *J. Chem. Phys.*, 143(5):054104, 2015.

[38] Daniel Gillespie. Approximate accelerated stochastic simulation of chemically reacting. *J. Chem. Phys.*, 115:1716–1733, 2001.

[39] Yang Cao, Daniel Gillespie, and Linda Petzold. Efficient step size selection for the tau-leaping simulation method. *J. Chem. Phys.*, 124(4):44109, 2006.

[40] Anne Auger, Philippe Chatelain, and Petros Koumoutsakos. R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *J. Chem. Phys.*, 125(8):84103, 2006.
[41] Vo H. Thanh, Roberto Zunino, and Corrado Priami. Accelerating rejection-based simulation of biochemical reactions with bounded acceptance probability. *J. Chem. Phys.*, 144(22):224108, 2016.

[42] Luca Marchetti, Corrado Priami, and Vo H. Thanh. HRSSA efficient hybrid stochastic simulation for spatially homogeneous biochemical reaction networks. *J. Comp. Phys.*, 317:301–317, 2016.

[43] Paola Lecca, Ian Laurenzi and Ferenc Jordan. Deterministic versus stochastic modelling in biochemistry and systems biology. *Woodhead Publishing Series in Biomedicine*, 2012.

[44] Muruhan Rathinam, Patrick W. Sheppard, and Mustafa Khammash. Efficient computation of parameter sensitivities of discrete stochastic chemical reaction networks. *J. Chem. Phys.*, 132(3):034103, 2010.

[45] David F. Anderson. An efficient finite difference method for parameter sensitivities of continuous time markov chains. *SIAM J. Numer. Anal.*, 50:2237–2258, 2012.

[46] Rishi Srivastava, David F. Anderson, and James B. Rawlings. Comparison of finite difference based methods to obtain sensitivities of stochastic chemical kinetic models. *J. Chem. Phys.*, 138(7):074110, 2013.

[47] Jacob A. McGill, Babatunde A. Ogunnaie and Dionisios G. Vlachos. Efficient gradient estimation using finite differencing and likelihood ratios for kinetic Monte Carlo simulations. *J. Comp. Phys.*, 231:7170–7186, 2012.

[48] Monjur Morshed, Brian Ingalls and Silvana Ilie. An efficient finite-difference strategy for sensitivity analysis of stochastic models of biochemical systems. *Biosystems*, 151:43–52, 2017.

[49] Sergey Plyasunov and Adam P. Arkin. Efficient stochastic sensitivity analysis of discrete event systems. *J. Comp. Phys.*, 221:724–738, 2007.

[50] Patrick B. Warren and Rosalind J. Allen. Steady-state parameter sensitivity in stochastic modeling via trajectory reweighting. *J. Chem. Phys.*, 136(10):104106, 2012.

[51] Patrick W. Sheppard, Muruhan Rathinam and Mustafa Khammash. A pathwise derivative approach to the computation of parameter sensitivities in discrete stochastic chemical systems. *J. Chem. Phys.*, 136(3):034115, 2012.

[52] Ankit Gupta and Mustafa Khammash. An efficient and unbiased method for sensitivity analysis of stochastic reaction networks. *J. R. Soc. Interface*, 11:20140979, 2014.

[53] Søren Asmussen and Peter W. Glynn. *Stochastic Simulation: Algorithms and Analysis. Springer*, 2007.
[54] Ramon E. Moore, R. Baker Kearfott, and Michael J. Cloud. *Introduction to Interval Analysis*. SIAM, 2009.

[55] Tobias Jahnke and Wilhelm Huisinga. Solving the chemical master equation for monomolecular reaction systems analytically. *Journal of Mathematical Biology*, 54(1):1–26, 2007.

[56] Luca Cardelli, Emmanuelle Caron, Philippa Gardner, Ozan Kahramanoğullari and Andrew Phillips. A Process Model of Rho GTP-binding Proteins. *Theoretical Computer Science*, 410:3166–3185, 2009.

[57] Ozan Kahramanoğullari and James Lynch. Stochastic Flux Analysis of Chemical Reaction Networks. *BMC Systems Biology*, 7:133, 2013.