Thanh, Vo Hong; Zunino, Roberto; Priami, Corrado

Efficient finite-difference method for computing sensitivities of biochemical reactions

Author for correspondence:
Vo Hong Thanh
e-mail: thanh.vo@aalto.fi

Efficient Finite Difference Method for Computing Sensitivities of Biochemical Reactions

Vo Hong Thanh[1], Roberto Zunino[2] and Corrado Priami[3]

[1] Aalto University, Department of Computer Science.
[2] University of Trento, Department of Mathematics.
[3] The Microsoft Research – University of Trento Centre for Computational and Systems Biology (COSBI), Italy and University of Pisa, Department of Computer Science

Sensitivity analysis of biochemical reactions aims at quantifying the dependence of the reaction dynamics on the reaction rates. The computation of the parameter sensitivities, however, poses many computational challenges when taking stochastic noise into account. This paper proposes a new finite difference method for efficiently computing sensitivities of biochemical reactions. We employ propensity bounds of reactions to couple the simulation of the nominal and perturbed processes. The exactness of the simulation is preserved by applying the rejection-based mechanism. For each simulation step, the nominal and perturbed processes under our coupling strategy are synchronized and often jump together, increasing their positive correlation and hence reducing the variance of the estimator. The distinctive feature of our approach in comparison with existing coupling approaches is that it only needs to maintain a single data structure storing propensity bounds of reactions during the simulation of the nominal and perturbed processes. Our approach allows to compute sensitivities of many reaction rates simultaneously. Moreover, the data structure does not require to be updated frequently, hence improving the computational cost. This feature is especially useful when applied to large reaction networks. We benchmark our method on biological reaction models to prove its applicability and efficiency.

1. Introduction

Biochemical reactions at cellular level are 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) can be amplified when key species like genes, mRNAs 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 [1–5] or cell fate decision [6]. The noise may be further propagated across cells leading to remarkable diversity at organism level [7,8].

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 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 infinitesimal time is assigned a probability which is proportional to a *propensity* function. The propensity of a reaction depends on the population numbers of 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 DM with improved search [12–14], with tree-based search [15–17], with composition-rejection search [18] and with partial-propensity approach [19], the next reaction method (NRM) [20,21], the rejection-based SSA (RSSA) [22–26] and other improvements [27–34]. Extensions of SSA have been introduced to cope with different aspects of biochemical reactions like time delays [14,21,35–37] and time-dependent reaction rates [21,38–40].

The dynamical behavior of biochemical 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 dynamics on their rates. Sensitivity analysis aims at quantitatively characterizing this dependency. Different methods for sensitivity analysis of stochastic chemical kinetics have been introduced including finite differences [41–43,45], likelihood ratios [44,46,47], infinitesimal perturbation analysis [48] and other [49]. Each of these methods has its own advantages and drawbacks (see Asmussen and Glynn [50] 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 their nominal rate values and the second one applies to perturbed values, respectively. The simulation requires two streams of independent random numbers, resulting in large variance of the crude estimator. To reduce the variance of the finite difference estimator, Rathinam et al. [41] 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. The correlation of the nominal and perturbed processed introduced by CRN, however, will be broken if the simulation time is large. Anderson [42] recently 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 processes with integrated propensities. We note that Rathinam et al. [41] also proposes a method that employs the random time change representation, but is often more involved than CFD. For each reaction, CFD splits the Poisson processes associated with the reaction in the nominal and perturbed processes so that a common Poisson process having smallest rate will be shared by these processes. CFD thus requires an additional computational cost for maintaining and sampling the shared Poisson processes. This extra computational cost is increasing with the number of reactions, which negatively affects the performance of the 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, called the rejection-based finite difference (RFD) method, to address the drawbacks of the state-of-the-art coupling strategies for sensitivity analysis of biochemical reactions. RFD constructs the estimator by using propensity bounds of reactions to couple the nominal and perturbed processes and employing the rejection-based simulation approach by Thanh et al. [22] to simulate these processes. The propensity bound of a reaction is an interval that encloses the propensity values of the reaction in both the nominal and perturbed processes. The propensity bound of each reaction is derived by bounding the population of reactant species and its rate values in these processes. Employing the propensity bounds of reactions and the rejection-based simulation allows RFD to correlate and synchronize the selection of reactions in the nominal and perturbed processes in each step. Specifically, for each simulation step, a reaction is selected as a candidate for firing in both the nominal and perturbed processes using its propensity bound. Knowing the candidate reaction, each process evaluates the exact propensity value of the candidate with its own state and decides whether to fire the candidate by applying the rejection-based test. The simulation of the nominal and perturbed processes often fire the candidate reaction and then jump their states together, increasing the positive correlation between them and hence reducing the variance of the estimator. The use of propensity bounds of reactions to implement the finite difference estimator offers many computational advantages in comparison with CRN [41] and CFD [42]. First, we only need a single data structure to store propensity bounds of reactions during the simulation of the nominal and perturbed processes. This enables RFD to compute sensitivities by simultaneously perturbing many reaction rates in a run, instead of one at a time as in CRN and CFD. We remark that the exactness of the marginal distributions of the processes by RFD is preserved by the application of the rejection-based mechanism. Second, our approach does not need to simulate and maintain additional processes as in CRN and CFD, which is often computationally expensive for large models. In addition, the propensity bounds of reactions in RFD are only updated infrequently. This computational efficiency of our approach makes it especially useful for large models.

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 scheme that uses the concept of propensity bounds and the rejection-based approach for computing sensitivities. The general framework for stochastic simulation of biochemical reactions using the rejection-based technique is recalled in Section (a). The description of the rejection-based coupling strategy is in Section (b) and its detailed implementation is outlined in Section (c). Section 4 presents the experimental results of the application of our approach to biological reactions 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 an n-vector $X(t) = (X_1(t), ..., X_n(t))$.

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.

$$R_j : v_{1j}S_1 + ... + v_{nj}S_n \xrightarrow{c_j} v'_{1j}S_1 + ... + v'_{nj}S_n \tag{2.1}$$
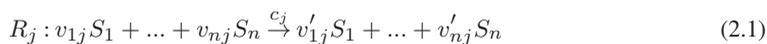
where the species on the left side of the arrow are called reactants and the ones on the right side are called products. The non-negative integer v_{ij} and v'_{ij}, called *stoichiometric coefficients*, denote the number of molecules of a reactant consumed and the number of a product produced by firing R_j, respectively. Each reaction is associated with a parameter c_j which is called the (stochastic) reaction rate constant.

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 denotes the change amount in the system state $X(t)$ caused by reaction R_j when it is selected to fire. The state change vector v_j of a

reaction R_j is an n-vector where the ith 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, the propensity a_j of reaction R_j is proportional its reactants and reaction rate constant c_j and can be computed as:

$$a_j(X(t)) = c_j h_j(X(t)) \tag{2.2}$$

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

The dynamics of state $X(t)$ under the stochastic chemical kinetics framework is modeled as a (continuous-time) jump Markov process and can be fully described by the chemical master equation (CME) [9]. The exact stochastic simulation algorithm (SSA) [10,11] can be applied to realize $X(t)$. 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_μ fires in the next infinitesimal time $t + \tau + d\tau$, given the state $X(t)$ at time t. Its closed form is:

$$p(\tau, \mu) = a_\mu exp(-a_0 \tau) \tag{2.3}$$

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

Various Monte Carlo strategies have been introduced for SSA in order to sample the pdf $p(\tau, \mu)$ [10,20, 22]. One approach is the direct method (DM) which samples the pdf $p(\tau, \mu)$ in Eq. 2.3 by using the fact that reaction R_μ fires with a discrete probability a_μ/a_0 and the firing time τ is exponentially distributed with rate a_0. Thus, for each simulation iteration, m propensities a_j for $j = 1 \dots m$ and their sum $a_0 = \sum_{j=1}^{m} a_j$ are computed. The next reaction firing R_μ with probability a_μ/a_0 is selected by

$$\mu = \text{smallest reaction index such that: } \sum_{j=1}^{\mu} a_j \geq r_1 a_0 \tag{2.4}$$

and the firing time τ is generated as

$$\tau = \frac{1}{a_0} \ln \left(\frac{1}{r_2} \right) \tag{2.5}$$

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 [20] 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.

An alternative approach to construct realizations of $X(t)$ is to use the next reaction method (NRM) by employing the random time change (RTC) representation [21]. Let Po_j be independent unit-rate Poisson processes denoting the number firings of reactions R_j, with $j = 1, \dots, m$. The dynamics of $X(t)$ under RTC can be expressed as

$$X(t) = X(0) + \sum_{j=1}^{m} Po_j \left(\int_0^t a_j(X(s))ds \right) v_j \tag{2.6}$$

Using the RTC representation, each reaction R_j has an internal time T_j, which determines the starting time of the clock until reaction R_j fires at the next time $t + \tau_j$, given the current time t. Let P_j be the next firing time of reaction R_j assuming that a_j is not changing, Then, the waiting time until its firing is $\tau_j = (1/a_j)(P_j - T_j)$. Because Po_j is a unit Poisson process, the next firing time P_j is generated by sampling an exponential distribution with rate 1. Having the waiting times of all reactions, the reaction R_μ having smallest waiting time $\tau = \min_{j=1}^{m}(\tau_j)$ is selected to fire. By explicitly keeping track of the firing times of Poisson processes, NRM consumes only one random number per simulation step.

(a) 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) [41] and coupled finite difference (CFD) [42]. These methods derive the sensitivity of the system dynamics by applying a small perturbation to a reaction rate constant. The computation of sensitivities with respect to several reaction rates requires the computation being performed for each rate separately.

Let c be an m-vector in which the jth element is the reaction rate constant c_j of a reaction R_j for $j = 1, \dots m$. 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:

$$S(c) = \mathbb{E}[f(X^c(t))] \tag{2.7}$$

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

Let R_k be the reaction for which we want to quantify the dependence of $S(c)$ on its reaction rate constant c_k. The aim 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 ϵ_k to the reaction rate c_k. Let e_k be a unit m-vector in which the kth 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*

$$\frac{\partial S(c)}{\partial c_k} \approx \frac{S(c + \epsilon_k e_k) - S(c)}{\epsilon_k} \approx \frac{\mathbb{E}[f(X^{c+\epsilon_k e_k}(t))] - \mathbb{E}[f(X^c(t))]}{\epsilon_k} \tag{2.8}$$

The bias of the forward difference due to the truncation error is $O(\epsilon_k)$. We note that the bias can be reduced to $O(\epsilon_k^2)$ by using the *centered finite difference* method [50]. In Eq. 2.8, the bias becomes zero in the limit that $\epsilon_k \to 0$.

The estimator for the forward difference in Eq. 2.8 can be constructed as

$$Z = \frac{1}{N} \sum_{i=1}^{N} \frac{f(X^{c+\epsilon_k e_k}_{[i]}(t)) - f(X^c_{[i]}(t))}{\epsilon_k} \tag{2.9}$$

where N is the number of simulation runs and $X^c_{[i]}(t)$ denotes the ith realization with rate parameter c. A naive implementation of the estimator where $X^c_{[i]}(t)$ and $X^{c+\epsilon_k e_k}_{[i]}(t)$ are generated independently will produce a large variance. In fact, the variance of the estimator $var[Z]$ in the naive implementation is equal the sum of two variances $var[f(X^{c+\epsilon_k e_k}(t))]$ and $var[f(X^c(t))]$ because their covariance is zero (i.e., $cov[f(X^{c+\epsilon_k e_k}(t)), f(X^c(t))] = 0$). 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 $cov[f(X^{c+\epsilon_k e_k}(t)), f(X^c(t))] \neq 0$, 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 the CRN approach as applied to SSA. The key of the CRN is that the random number 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.

Algorithm 1: Common Random Number method (CRN)

1: define reaction index k for sensitivity analysis
2: initialize time $t = 0$ and state $X^c = X^{c+\epsilon_k e_k} = x_0$
3: generate a random seed w
4: seed the random number generator with w
5: realize X^c by performing SSA until time T_{max}
6: reseed the random number generator with w
7: realize $X^{c+\epsilon_k e_k}$ by performing SSA until time T_{max}

The CFD employs the RTC representation in Eq. 2.6 to correlate the nominal process $X^c(t)$ and its perturbed one $X^{c+\epsilon_k e_k}(t)$. Specifically, by RTC representation and additive property of the Poisson process, it gives

$$X^c(t) = X(0) + \sum_{j=1}^{m} Po_{j,1}\left(\int_0^t b_j(s)ds\right)v_j + \sum_{j=1}^{m} Po_{j,2}\left(\int_0^t (a_j(X^c(s)) - b_j(s))ds\right)v_j \quad (2.10)$$

and

$$X^{c+\epsilon_k e_k}(t) = X(0) + \sum_{j=1}^{m} Po_{j,1}\left(\int_0^t b_j(s)ds\right)v_j + \sum_{j=1}^{m} Po_{j,3}\left(\int_0^t (a_j(X^{c+\epsilon_k e_k}(s)) - b_j(s))ds\right)v_j$$
$$(2.11)$$

where $b_j(t) = \min(a_j(X^c(t)), a_j(X^{c+\epsilon_k e_k}(t)))$ and $Po_{j,i}$ 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. 2.10 - 2.11 is the key of the CFD method outlined in Algorithm 2 where the simulation of the Poisson processes $Po_{j,1}(\int_0^t b_j(s)ds)$ is shared.

Algorithm 2: Coupled Finite Difference method (CFD)

1: define reaction index k for sensitivity analysis
2: initialize time $t = 0$ and state $X^c = X^{c+\epsilon_k e_k} = x_0$
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** $(t < T_{max})$ **do**
5:     compute $a_j(X^{c+\epsilon_k e_k})$ and $a_j(X^c(t))$ for $j = 1, \ldots, m$
6:     set $b_{j,1} = \min(a_j(X^{c+\epsilon_k e_k}), a_j(X^c(t)))$ and compute $b_{j,2} = a_j(X^c) - b_{j,1}$ and $b_{j,3} = a_j(X^{c+\epsilon_k e_k}) - b_{j,1}$ 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 (μ, α) be the pair of indices where the minimum is selected
9:     set $t = t + \tau$
10:     **if** $(\alpha == 1)$ **then**
11:       update $(X^c, X^{c+\epsilon_k e_k}) = (X^c, X^{c+\epsilon_k e_k}) + (v_\mu, v_\mu)$
12:     **else if** $(\alpha == 2)$ **then**
13:       update $X^c = X^c + v_\mu$
14:     **else if** $(\alpha == 3)$ **then**
15:       update $X^{c+\epsilon_k e_k} = X^{c+\epsilon_k e_k} + v_\mu$
16:     **end if**
17:     set $T_{j,i} = T_{j,i} + b_{j,i}\tau$ for $j = 1, \ldots, m$ and $i = \{1, 2, 3\}$
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 scheme that employs the concept of propensity bounds and rejection-based simulation technique for efficiently computing sensitivities of biochemical reactions. We first recall the principle of the rejection-based stochastic simulation algorithm (RSSA), which provides the theoretical framework for our new coupling strategy. Then, we describe in detail how our new rejection-based finite difference (RFD) method correlates the simulation of the nominal and perturbed processes to construct the estimator. We also compare our coupling strategy with the CRN and CFD methods discussed in previous section to highlight the advantages of our approach.

(a) Background on rejection-based simulation

RSSA, originally proposed by Thanh *et al.* [22], is an exact stochastic simulation algorithm that aims to improve simulation performance by reducing the average number of propensity calculations during the simulation. The exactness proof of RSSA as well as its extensions for handling complex reaction mechanisms can be accessed through the recent work [14,22,31,39,40]. The mathematical framework for the selection of reaction firings in RSSA is the rejection-based sampling technique. It uses propensity bounds $[\underline{a_j}, \overline{a_j}]$, 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_μ is selected with probability $\overline{a_\mu}/\overline{a_0}$ where $\overline{a_0} = \sum_{j=1}^m \overline{a_j}$. Second, the candidate R_μ is validated through a rejection test with success probability $a_\mu(X(t))/\overline{a_\mu}$. If the candidate R_μ 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_μ is accepted with probability $\underline{a_\mu}/\overline{a_\mu}$, then it can be accepted without evaluating $a_\mu(X(t))$ because of $\underline{a_\mu}/\overline{a_\mu} < a_\mu(X(t))/\overline{a_\mu}$.

RSSA also uses the propensity bounds of reactions to compute the firing time of the accepted candidate reaction. Let l be the number of trials such that the first $l - 1$ trials are rejections and at the lth trial R_μ is accepted. The firing time τ of the accepted candidate R_μ in RSSA is the sum of l independent exponentially distributed numbers with the same rate $\overline{a_0}$, which is equivalent to an `Erlang`$(l, \overline{a_0})$ distribution. RSSA thus generates the firing time τ of the accepted candidate R_μ by sampling the corresponding *Erlang* distribution.

(b) Coupling by rejection-based approach

We describe in this section the principle of the rejection-based finite difference (RFD) method that employs the rejection-based framework introduced by RSSA for coupling the simulation of the original and perturbed processes, with the aim of computing sensitivities. Let k be the index of reaction R_k, for which we want to measure the sensitivity. Let $X^c(t)$ be the state of the nominal process and $X^{c+\epsilon_k e_k}(t)$ be state of the perturbed process obtained by changing kth element of the rate vector c by an amount denoted by $\epsilon_k e_k$. Consider a time interval $[0, t]$. Let $\overline{a_j}$ be an arbitrary propensity upper bound that is greater than the exact propensity values $a_j(X^c(t))$ and $a_j(X^{c+\epsilon_k e_k}(t))$ of each reaction R_j, with $j = 1, \ldots, m$, during the time interval, i.e., $\overline{a_j} \geq a_j(X^c(s))$ and $a_j(X^{c+\epsilon_k e_k}(s))$ for all $s \in [0, t]$. Let Po_j be a unit-rate Poisson process, evaluated over the time interval $[0, t]$. By the addictive property of the Poisson distribution, the Poisson process with rate $\overline{a_j}$ can be decomposed as

$$Po_j\left(\overline{a_j}t\right) = Po_{j,1}\left(\int_0^t a_j(X^c(s))ds\right) + Po_{j,2}\left(\overline{a_j}t - \int_0^t a_j(X^c(s))ds\right) \tag{3.1}$$

and

$$Po_j\left(\overline{a_j}t\right) = Po_{j,3}\left(\int_0^t a_j(X^{c+\epsilon_k e_k}(s))ds\right) + Po_{j,4}\left(\overline{a_j}t - \int_0^t a_j(X^{c+\epsilon_k e_k}(s))ds\right) \tag{3.2}$$

Equivalently, these equations can be rewritten as

$$Po_{j,1}\left(\int_0^t a_j(X^c(s))ds\right) = Po_j\left(\overline{a_j}t\right) - Po_{j,2}\left(\overline{a_j}t - \int_0^t a_j(X^c(s))ds\right) \tag{3.3}$$

and

$$Po_{j,3}\left(\int_0^t a_j(X^{c+\epsilon_k e_k}(s))ds\right) = Po_j\left(\overline{a_j}t\right) - Po_{j,4}\left(\overline{a_j}t - \int_0^t a_j(X^{c+\epsilon_k e_k}(s))ds\right) \tag{3.4}$$

Thus, by the RTC representation, the resulting dynamics of the nominal process is

$$X^c(t) = X(0) + \sum_{j=1}^{m} \left[Po_j\left(\overline{a}_j t\right) - Po_{j,2}\left(\overline{a}_j t - \int_0^t a_j(X^c(s))ds\right) \right] v_j \tag{3.5}$$

and its perturbed one is

$$X^{c+\epsilon_k e_k}(t) = X(0) + \sum_{j=1}^{m} \left[Po_j\left(\overline{a}_j t\right) - Po_{j,4}(\overline{a}_j t - \int_0^t a_j(X^{c+\epsilon_k e_k})(s)ds) \right] v_j \tag{3.6}$$

We remark that the equality in these derivations is used in the sense of having the same distribution.

Eqs. 3.5 and 3.6 give the mathematical basis for our coupling strategy. These equations show that if we simulate the nominal process and the perturbed process using the propensity bounds while filtering out selected candidate reactions using the corresponding exact propensities, then we can positively couple the simulation of these processes. This is made possible by the rejection-based simulation technique. We note that the original rejection-based selection in RSSA, however, does not directly apply, because there each trajectory is generated using independent random numbers, causing no correlation between them. To make a concrete example, when using RSSA it is possible that the nominal process accepts a reaction firing after two trials, while the perturbed process only accepts the firing after four trials, whose time points are even different with respect to those for the nominal process. Instead, for computing sensitivities we need to improve the correlation, hence we need to synchronize the nominal and perturbed processes in each trial. To do this, the key idea underlying RFD is to decompose the complex rejection-based selection of RSSA into single trials and assign a time stamp for each trial. In particular, for each simulation step, a candidate reaction R_j for both the processes is selected with a probability proportional to its propensity upper bound $\overline{a}_j$ and its waiting time is generated following an exponential distribution with rate $\overline{a}_0$. The rejection-based test then decides whether the candidate reaction will be accepted to fire in the corresponding process with the exact propensity $a_j(X^c(t))$ and $a_j(X^{c+\epsilon_k e_k}(t))$, respectively. Furthermore, by choosing appropriate propensity bounds for the simulation, we can make the candidate reaction to be accepted to fire in both processes frequently during the simulation, moving the states together, hence positively correlating these processes. The derivation of propensity bounds as well as the implementation of RFD are discussed in the next section. In the following, we analyze the efficiency of our rejection-based coupling strategy and compare it with CRN and CFD.

The application of the rejection-based coupling strategy represented in Eqs. 3.5 and 3.6 can reduce the variance of the estimator. Specifically, by subtracting $X^{c+\epsilon_k e_k}(t)$ in Eq. 3.5 and $X^{c+\epsilon_k e_k}(t)$ in Eq. 3.6 then taking variance of the result, it shows that the variance of the estimator is proportional to the variance of Poisson processes having rate equal to the difference of the propensity upper bound and the exact propensity value. The propensity upper bounds of reactions, however, can be chosen with flexibility in which it provides us possibility for tuning RFD to improve both the variance of the estimator and the simulation performance. In particular, if we choose $\overline{a}_j = \max(a_j(X^c(s)), a_j(X^{c+\epsilon_k e_k}(s)))$, then our RFD approach produces an estimator that has the same variance as CFD represented in Eqs. 2.10 and 2.11, which use the minimum value of $a_j(X^c(s))$ and $a_j(X^{c+\epsilon_k e_k}(s))$ to couple these processes. However, by defining the propensity bounds in this way, we have to update these bounds after each simulation step in order to maintain the constraint, hence decreasing the simulation performance. The implementation of RFD relaxes this constraint by deriving the propensity upper bounds $\overline{a}_j$ such that they can be used for many steps without the need to update, thus improving the simulation performance. We remark that the simulation results are ensured to be exact, regardless of the choice of the propensity bounds due to the rejection-based mechanism.

Although our rejection-based coupling in Eqs. 3.5 and 3.6 involves the additional Poisson processes with rates equal to the differences of the propensity upper bound and the exact propensity values (i.e, $Po_{j,2}(\overline{a}_j t - \int_0^t a_j(X^c(s))ds)$ and $Po_{j,4}(\overline{a}_j t - \int_0^t a_j(X^{c+\epsilon_k e_k}(s))ds)$), we do not simulate all these processes explicitly during the simulation. In fact, these residual processes are only evaluated when some candidate is accepted. This is the distinctive feature of our RFD approach, compared with the CRN and CFD coupling strategies, which either have to simulate both the nominal and perturbed processes separately, as in CRN, or require to explicitly simulate and maintain additional Poisson processes, as in CFD. We remark that the additional computational cost introduced by the CRN and CFD methods increases with

the number of reactions in the network, which negatively affects the computational performance of these approaches when simulating large models. Our rejection-based coupling only requires to keep track of propensity bounds of reactions, which are also updated infrequently. Furthermore, the use of propensity bounds for coupling allows to simulate many distinct perturbations on rates in a single run. In this case, we only need to define a propensity upper bound that constrains the propensities of reactions for all these perturbations. Having the propensity bounds, the same rejection-based coupling strategy can be applied to couple and simulate these processes.

(c) RFD algorithm

We outline in Algorithm 3 the detailed implementation of RFD for realizing the nominal and perturbed processes used in sensitivity analysis. It takes a set $\mathcal{Q}$ containing reaction indices for which sensitivities need to be computed. The algorithm will generate realizations of X^c and $X^{c+\epsilon_k e_k}$ for reaction $k \in \mathcal{Q}$ in which $\epsilon_k e_k$ denotes the amount of change in the rate of the reaction index k. The simulation starts with an initial state x_0 at time $t = 0$ and ends at time T_{max}.

For the initialization in lines 3 - 7, RFD computes propensity bounds $\underline{a_j}$ and $\overline{a_j}$ for each reaction R_j, $j = 1, \ldots, m$, such that $\underline{a_j} \leq a_j(X^c(t)), \{a_j(X^{c+\epsilon_k e_k}(t))\}_{k \in \mathcal{Q}} \leq \overline{a_j}$. The propensity upper bounds $\overline{a_j}$ will be used to couple the simulation of the nominal and the perturbed processes, while the propensity bounds are used to quickly accept the candidate reaction, hence avoiding computing the exact propensity value and improving the simulation performance.

RFD computes the propensity bound $[\underline{a_j}, \overline{a_j}]$ for each reaction $R_j, j = 1, \ldots, m$, by bounding the rates of the reaction R_j in both the nominal and perturbed processes as well as constraining all populations of reactant species in $X^c(t)$ and $X^{c+\epsilon_k e_k}(t)$ with $k \in \mathcal{Q}$ into the fluctuation interval $[\underline{X}, \overline{X}]$.

For each reaction R_j, RFD defines a lower value $\underline{c_j}$ 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 $\underline{c_j} = \min(c_j, c_j + \epsilon_j)$ 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 $\underline{c_j} = \overline{c_j} = c_j$.

The derivation of the lower bound $\underline{X_i}$ 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 $X^{c+\epsilon_k e_k}(t)$ for $k \in \mathcal{Q}$, hence forming the fluctuation interval $[\underline{X}, \overline{X}]$, is obtained by constraining the minimum and maximum population of this species in all these states. Precisely, RFD defines $X_i^{min} = \min(X_i^c(t), \{X_i^{c+\epsilon_k e_k}(t)\}_{k \in \mathcal{Q}})$ and $X_i^{max} = \max(X_i^c(t), \{X_i^{c+\epsilon_k e_k}(t)\}_{k \in \mathcal{Q}})$ as the minimum and maximum population of species S_i, respectively. The computation of the population bounds for S_i is thus $\underline{X_i} = (1 - \delta_i)X_i^{min}$ and $\overline{X_i} = (1 + \delta_i)X_i^{max}$ where the *fluctuation rate* $0 \leq \delta_i \leq 1$ is a parameter. The choice of the fluctuation rate δ_i is the trade-off between reducing the update costs and increasing rejections, both affecting the simulation performance. If δ_i is too big ($\delta_i \approx 1$), we do not need to update propensity bounds often; however, it also increases the number of rejections, negatively affecting simulation performance. On the other hand, if δ_i is too small ($\delta_i \approx 0$), then we need to recompute the propensity bounds frequently, making the simulation inefficient. For typical models, the fluctuation rate chosen around 10% to 20% gives a good simulation performance (see numerical examples in Section 5). We note that the choice of the fluctuation rate δ_i, however, does not affect the simulation result, which is always exact due to the rejection-based mechanism..

The propensity lower bound $\underline{a_j}$ 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 $[\underline{c_j}, \overline{c_j}]$ and fluctuation interval $[\underline{X}, \overline{X}]$. For mass-action propensity function a_j given in Eq. 2.2, the bounds can be computed easily using its monotonic property and interval arithmetic [51]. Specifically, let $\underline{h_j}$ and $\overline{h_j}$ be the minimum and maximum of function h_j over the fluctuation interval $[\underline{X}, \overline{X}]$, respectively. By monotonic property of function h_j, it gives $\underline{h_j} = h_j(\underline{X})$ and $\overline{h_j} = h_j(\overline{X})$. Then by interval analysis, the propensity bounds of R_j can be computed as

$$\underline{a_j} = \underline{c_j}\underline{h_j} \tag{3.7}$$

and

$$\overline{a_j} = \overline{c_j}\overline{h_j} \tag{3.8}$$

The main simulation loop of the RFD algorithm using propensity bounds $[\underline{a_j}, \overline{a_j}]$ of reactions is composed of two main tasks. The first task selects the reaction to update the states using propensity bounds

---

Algorithm 3: Rejection-based Finite difference method (RFD)

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

---

and the rejection-based mechanism (lines 10 - 29). The second task updates propensity bounds of reactions when there exists a species whose population moves out of the current fluctuation interval (lines 30 - 40). In addition, to facilitate the 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 [22] to retrieve which reactions should update their propensity bounds when a species exits its fluctuation interval. The SR

dependency graph $\mathcal{G}$ 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 $\mathcal{G}$ is built in line 2.

For the selection step in lines 10 - 29, a reaction firing is selected to update the states X^c or $X^{c+\epsilon_k e_k}$ for $k \in \mathcal{Q}$ by the rejection-based selection and taking three random numbers r_1, r_2 and $r_3 \sim U(0, 1)$ in which r_1 are used to compute time, while r_2 and r_3 is used to select the candidate and to validate it through a rejection test as follows. Let $\overline{a_0} = \sum_{j=1}^m \overline{a_j}$ be the sum of propensity upper bounds. The waiting time τ of in single trial of the rejection-based selection is an exponential distribution with rate $\overline{a_0}$. It thus can be computed by the inverse transformation as $\tau = (1/\overline{a_0}) \ln(1/r_1)$. The selection of reaction firing is composed of two steps. First, candidate reaction R_μ is randomly selected with a discrete probability $\overline{a_\mu}/\overline{a_0}$. The realization of the candidate reaction R_μ can be performed by linearly accumulating propensity upper bounds until it finds the smallest reaction index μ satisfying the inequality: $\sum_{j=1}^\mu \overline{a_j} > r_2 \cdot \overline{a_0}$ where $r_2 \sim U(0, 1)$. Knowing the candidate reaction R_μ, RFD applies the rejection-based test to decide whether to update the states depending on the value of the propensity of the reaction in the corresponding process. 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, RFD first checks whether $r_3 \leq \underline{a_\mu}/\overline{a_\mu}$ holds. If in fact this is the case, then both the states $X^c(t)$ and $X^{c+\epsilon_k e_k}$, for $k \in \mathcal{Q}$, are updated. If this test fails, RFD computes the propensities of R_μ 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 $a_\mu(X^{c+\epsilon_k e_k}(t))$ for each $k \in \mathcal{Q}$. Then, it checks whether $r_3 \leq a_\mu(X^c(t))/\overline{a_\mu}$ (respectively, $r_3 \leq a_\mu(X^{c+\epsilon_k e_k}(t))/\overline{a_\mu}$) in order to update $X^c(t)$ (respectively, $X^{c+\epsilon_k e_k}(t)$).

We emphasize that the selection of reaction firing in lines 10 - 29 produces the exact marginal distribution for states $X^c(t)$ and $X^{c+\epsilon_k e_k}(t)$ for $k \in \mathcal{Q}$. A sketch of the correctness proof is as follows. We have for each selection step a candidate reaction R_μ is selected with probability $\overline{a_\mu}/\overline{a_0}$ and its waiting time is exponentially distributed with rate $\overline{a_0}$. Now consider the particular state $X^{c+\epsilon_k e_k}(t)$. The accepted probability of R_μ is $a_\mu(X^{c+\epsilon_k e_k}(t))/\overline{a_\mu}$. The probability that R_μ is selected and accepted is thus $a_\mu(X^{c+\epsilon_k e_k}(t))/\overline{a_0}$. Let l_k be the number of trials until a candidate reaction is accepted to update the state $X^{c+\epsilon_k e_k}(t)$. The firing time at which the accepted candidate R_μ updates state $X^{c+\epsilon_k e_k}(t)$ is the sum of l_k exponential random numbers, thus following the `Erlang`$(l_k, \overline{a_0})$ distribution. From this point, the correctness argument of RSSA [22] can be adapted to prove R_μ selected correctly with probability proportional to $a_\mu(X^{c+\epsilon_k e_k}(t))$, which ensures the exact marginal distribution of $X^{c+\epsilon_k e_k}(t)$.

The selection is repeated 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 the propensity bounds of reactions should be updated. Line 30 - 40 performs the update of propensity bounds when there existing species move out of their fluctuation interval. RFD keeps track of species that should update their fluctuation intervals by the set UpdateSpeciesSet, which is initialized to be an empty set at the beginning in line 9. For each species $S_i \in$ UpdateSpeciesSet, a new fluctuation interval $[\underline{X_i}, \overline{X_i}]$ that bounds all populations of the species in all states is computed. Then, reactions affected by S_i, denoted by the set ReactionsAffectedBy(S_i), is extracted from SR graph $\mathcal{G}$. For each $R_j \in$ ReactionsAffectedBy(S_i), its new lower bound $\underline{a_j}$ 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 the implementations of CRN and CFD, we use the dependency graph [20] 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.

(a) 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. 4.1.

$$\varnothing \xrightarrow{c_1} S \xrightarrow{c_2} \varnothing \qquad (4.1)$$

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 $a_2 = c_2 \# S$. For this model, we focus on sensitivities of the population of species S at time t due to reaction rates.

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 [52] as the sum of Binomial distribution $Bin(s_0, p)$ and Poisson distribution $Po(\lambda)$ 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

$$\mathbb{E}[\# S(t)] = s_0 e^{-c_2 t} + (c_1/c_2)(1 - e^{-c_2 t}) \qquad (4.2)$$

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 $T_{max} = 100$. First, we compute the sensitivities of the population of species S by increasing the reaction rate constant c_2 an amount of $\epsilon_2 = 10\%$ of this rate constant. We remark that the perturbation size in this section is defined as the percentage of change rather than the absolute value. The percentage is used in order to normalize the perturbation sizes [44]. For the simulation of RFD, the fluctuation rate 10% is applied to compute the fluctuation interval of species S. 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 the variances obtained by CFD and RFD are better than CRN.
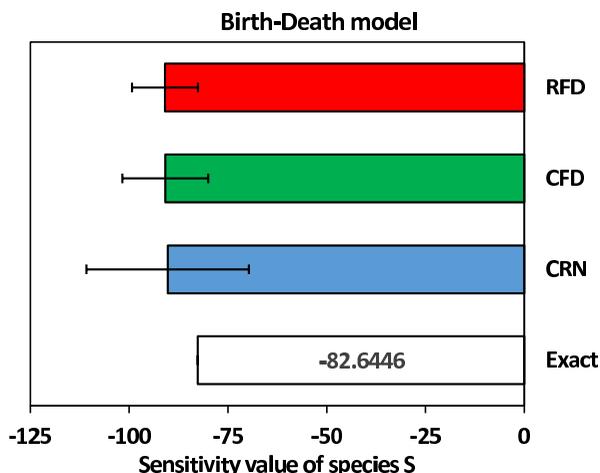

Figure 1: Sensitivities of the expected value of population of species S by CRN, CFD and RFD methods in comparison with exact value. The nominal reaction rate constant c_2 is increased by $\epsilon_2 = 10\%$. The sensitivity values are obtained by 1000 runs of these methods each with simulation time $T_{max} = 100$.

Figure 3 shows performance of CRN, CFD and RFD in computing the sensitivities of the population of species S by increasing the reaction rate constant 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 reducing 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 4.0×10^4 and by CFD is 2.06×10^4. RFD performs 2.21×10^4 simulation steps, but only has to update propensity bounds about 160 times.
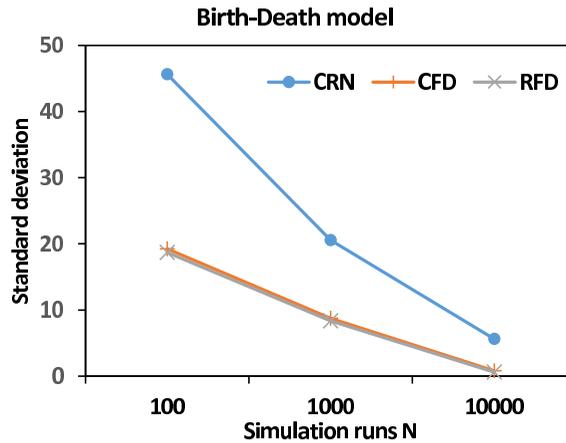
Figure 2: Standard deviations in estimating sensitivities of the expected value of population of species S by perturbing the reaction rate constant c_2 by an amount $\epsilon_2 = 10\%$ using CRN, CFD and RFD methods with different number of simulation runs.
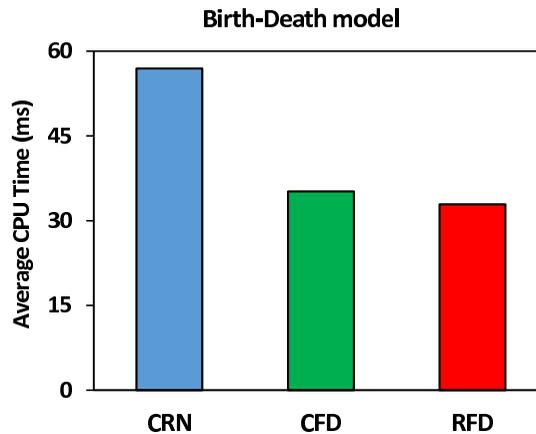

Figure 3: Performance of CRN, CFD and RFD methods in computing sensitivities of species S where the nominal reaction rate constant c_2 is increased by $\epsilon_2 = 10\%$.

Table 1 shows the effect of the fluctuation rate, hence the values of propensity bounds, to the estimated sensitivity and performance of RFD. The results are obtained by 1000 runs on the birth death process where reaction rate constant c_2 is increased by $\epsilon_2 = 10\%$. The table shows that the choice of fluctuation rate does not affect the sensitivity estimated by RFD, but only its performance. We note that the performance of RFD when the fluctuation rate is very small or very large is the worst. The small fluctuation rate, hence tight propensity bounds, causes many updates. The large fluctuation rate, hence loose propensity bounds, leads to many rejections. Both cases negatively affect the performance of RFD. In this experiment, the fluctuation rate 10% gives the best performance.

For the second experiment, we repeat the sensitivity analysis of the population of species S by simultaneously perturbing both the reaction rate constant c_1 by $\epsilon_1 = 1\%$ and the reaction rate constant c_2 by $\epsilon_2 = 10\%$. For the simulation of RFD, the fluctuation rate 10% is applied to compute the fluctuation interval of species S. The performance of algorithms 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.

Table 1: Sensitivity and performance of RFD with varying fluctuation rate in simulating the birth death process where reaction rate constant c_2 is increased by $\epsilon_2 = 10\%$.

Fluctuation rate	Sensitivity of S	Average CPU Time (ms)
1%	-90.18 ± 8.242	48.39
10%	-90.05 ± 8.345	34.65
20%	-90.06 ± 8.021	37.76
30%	-90.14 ± 8.180	37.99
90%	-90.03 ± 8.438	42.12

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 2.25×10^4 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.
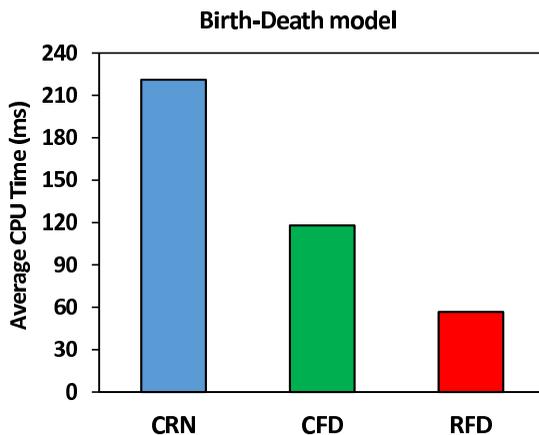

Figure 4: Performance of CRN, CFD and RFD methods in computing sensitivities of species S where both the reaction rate constant c_1 is increased by $\epsilon_1 = 1\%$ and the reaction rate constant c_2 is increased by $\epsilon_2 = 10\%$. The CRN and CFD methods have to repeat the computation 4 times corresponding to 4 combinations of perturbation parameters, while RFD performs simultaneously in one run.

(b) Rho GTP-binding protein model

We use the model of Rho GTP-binding proteins [53,54] 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 2: 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	$c_{23} = 0.02$

Table 2 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 $\epsilon_1 = 20\%$ and Figure 6 depicts the performance of CRN, CFD and RFD. The sensitivity values shown in Figure 5 are obtained by performing 1000 runs of algorithms with simulation time $T_{max} = 10$. These figures show that RFD is more efficient than both CFD and CRN in both the estimation of the sensitivities as well as performance.

In Figure 5, the sensitivity values of species RE and RD estimated by CRN are less reliable due to the loose coupling of processes during the simulation although the reaction rate constant c_1 is less sensitive. The result is that the sensitivities estimated by CRN vary significantly. CFD and RFD helps to solve the problem efficiently by employing tightly coupling strategies. The nominal and perturbed processes by CFD and RFD jump together almost of the time during the simulation which result in a more reliable estimation by these algorithms.

The performance plot in Figure 6 shows that RFD has the best performance, while CFD is the worst. The reason for the low performance of CFD in this experiment is due to the high computational cost for updating of propensities and related data structures of additional processes after reaction firings, even though CFD only performs 1.45×10^5 simulation steps which are a half as comparing with 2.88×10^5 steps by CRN and 2.53×10^5 steps by RFD, respectively. By reducing the propensity updates during the simulation, RFD

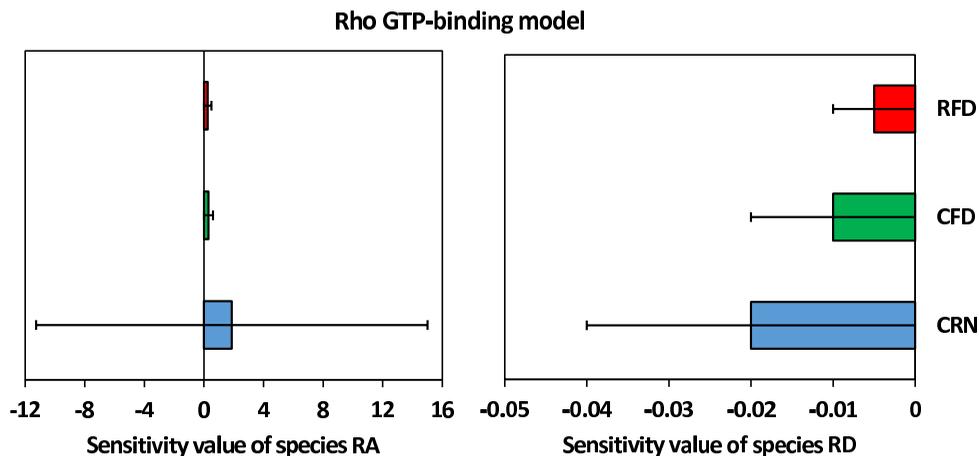Figure 5: Sensitivities of the expected value of population of species RA and RD by CRN, CFD and RFD methods. The nominal reaction rate constant c_1 is increased by $\epsilon_1 = 20\%$. The sensitivity values are obtained by 1000 runs of these methods each with simulation time $T_{max} = 10$.
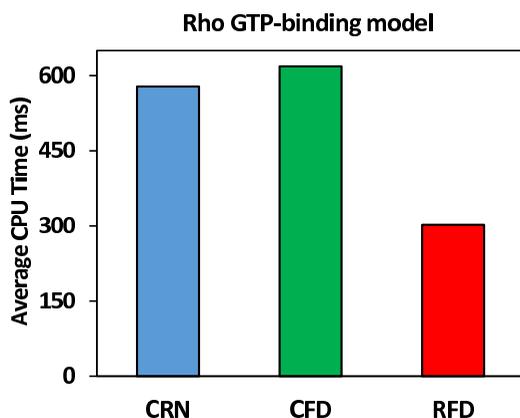

Figure 6: Performance of CRN, CFD and RFD methods in computing sensitivities of the expected value of population of species RA and RD where the reaction rate constant c_1 is increased by $\epsilon_1 = 20\%$.

significantly improves the performance. Specifically RFD only performs 2.13×10^4 propensity updates (about 9% of its simulation steps). The simulation 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 the rejection-based mechanism to construct the estimator. By employing propensity bounds of reactions, the simulations of reactions with nominal and perturbed rates can be positively correlated, hence reducing of the variance of the estimator. The exactness of the simulation is preserved by applying the rejection-based mechanism. The advantage of using propensity bounds for coupling allows our method to simultaneously perturb many reaction rates at a time. The computational gain of our method is achieved by reducing the propensity updates during the simulation. Our rejection-based coupling is thus very promising for further investigation such as performing sensitivity analysis of reactions with time-dependent rates or computing high-order sensitivities used for

optimization of biological processes. The weakness of our proposed rejection-based coupling is that the correlation may be loose if many simultaneous perturbations are applied and the dynamics of the nominal and perturbed processes significantly diverge. In such case, the propensity bounds will be large; thus, performance will suffer. In future work, we would investigate new strategies to mitigate the problem. It would also be interesting to understand whether the coupling is also affected.

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. 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.
13. 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.
14. 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.
15. James Blue, Isabel Beichl, and Francis Sullivan. Faster monte carlo simulations. *Phys. Rev. E*, 51(2):867–868, 1995.
16. Vo H. Thanh and Roberto Zunino. Tree-based search for stochastic simulation algorithm. In *Proc. of ACM-SAC*, pages 1415–1416, 2012.
17. 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.

18. 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.

19. Rajesh Ramaswamy, Nélido González-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.

20. 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.

21. 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.

22. 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.

23. 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.

24. Vo H. Thanh. A Critical Comparison of Rejection-Based Algorithms for Simulation of Large Biochemical Reaction Networks. *Bull Math Biol*, 2018. https://doi.org/10.1007/s11538-018-0462-y..

25. 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.*, 13(3):657–667, 2018.

26. 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/, 2013.

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

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

29. Anne Auger, Philippe Chatelain, and Petros Koumoutsakos. R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *J. Chem. Phys.*, 125(8):84103, 2006.

30. 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.

31. 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.

32. 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.

33. 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.

34. Luca Marchetti, Corrado Priami, and Vo H. Thanh. Simulation Algorithms for Computational Systems Biology. *Springer*, 2017.

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

36. 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.

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

38. 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.

39. 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.

40. Vo H. Thanh, Luca Marchetti, Federico Reali and Corrado Priami Incorporating extrinsic noise into the stochastic simulation of biochemical reactions: A comparison of approaches. *J. Chem. Phys.*, 148(6):064111, 2018.

41. 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.

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

43. 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.

44. Jacob A. McGill, Babatunde A. Ogunnaike 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.

45. 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.

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

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

48. 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.

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

50. Søren Asmussen and Peter W. Glynn. Stochastic Simulation: Algorithms and Analysis. *Springer*, 2007.

51. Ramon E. Moore, R. Baker Kearfott, and Michael J. Cloud Introduction to Interval Analysis. *SIAM*, 2009.

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

53. 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.

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