

An effective stepsize selection procedure for discrete simulation of biochemical reaction systems

Tianhai Tian¹

Kevin Burrage²

(Received 1 September 2006; revised 1 April 2008)

Abstract

We present an effective procedure for selecting the stepsize in the binomial τ -leap method, which is an efficient technique for the discrete simulation of biochemical reaction systems. We use the difference of the propensity functions to approximate their derivatives, thus giving a derivative-free implementation. We compare the difference between the stepsizes obtained by existing procedures and the new procedure, and compare their relative efficiencies when simulating biochemical reaction systems. Numerical results indicate that the new procedure is very efficient and robust. More importantly, this new procedure is easy to implement and leads one step further towards a general purpose computer program for the efficient simulation of stochastic biochemical reaction systems.

Contents

1 Introduction	C1023
2 Stochastic simulation algorithms	C1025
3 Stepsize selection procedures	C1028
4 Simulation results	C1032
5 Conclusion	C1036
References	C1037

1 Introduction

Stochastic modelling of biological systems has become a very important research field in system biology in recent years. Experimental and theoretical studies show the importance of stochastic processes in genetic regulatory networks and cellular processes. For biological systems involving molecules of small populations, the stochastic simulation algorithm (SSA) derived by Gillespie [4] is an essentially exact procedure for studying noise in biological systems. However, the computational time of the SSA is often very large when it is applied to simulate large scale biological systems. It is imperative to design efficient numerical methods for simulating stochastic chemical kinetics.

There are two significant approaches for reducing the computational time of the SSA. The first approach is based on a new technique of Gillespie through the use of leap methods with Poisson random variables [5]. In the Poisson τ -leap method, a number of reactions are allowed to fire in a relatively larger time interval rather than one single reaction firing in the next

reaction time interval, as is the case of the SSA. Following the Poisson τ -leap method, the midpoint τ -leap method [5], implicit τ -leap method [11], Poisson Runge–Kutta methods [2], and binomial τ -leap method [12] have been designed in order to improve the accuracy and efficiency of the stochastic simulation. In addition, an improved leap size selection procedure has been proposed for determining the maximum leap size for a specified degree of accuracy [6]. More sophisticated sampling techniques need to be developed in this approach in order to avoid obtaining negative molecular numbers in the application of the τ -leap methods.

The second approach is to partition a chemical reaction system into subsets of slow and fast reactions and then to apply different simulation methods to each subset. Rao and Arkin [10] demonstrated how to reduce computational time by applying the quasi-steady state assumption to the subset of fast reactions. Haseltine and Rawlings [7] improved the computational efficiency by approximating fast reactions either deterministically or as Langevin equations. Burrage et al. [3] partitioned chemical reaction systems into three subsets of slow, intermediate and fast reactions and used the Poisson τ -leap method to simulate the subset of intermediate reactions. In addition, Puchalka and Kierzek [9] studied the maximal time step method that is a combination of the SSA in Gibson and Bruck’s form and the Poisson τ -leap method. One of the challenging problems in this approach is the complexity of the partitioning process that may erode potential efficiency gains.

The ultimate goal of our research in this field is to develop a general-purpose computer program to simulate biochemical reaction systems based on the binomial τ -leap method. To reach this target, this work will study the procedure for selecting stepsizes in the τ -leap methods. Existing procedures are based on the calculation of derivatives of the propensity functions. Here we develop a simple procedure that is very efficient and easy to implement.

2 Stochastic simulation algorithms

When applying stochastic simulation algorithms to simulate biochemical reaction systems, a well stirred chemical reaction system contains N molecular species $\{S_1, \dots, S_N\}$ with number $x_i(t)$ of the species S_i at time t . These species of molecules chemically interact inside some fixed volume Ω at a constant temperature through reaction channels $\{R_1, \dots, R_M\}$.

For each reaction channel R_j ($j = 1, \dots, M$), we define a propensity function $a_j(\mathbf{x})$ in a given state $\mathbf{x} = (x_1(t), \dots, x_N(t))^T$ and use $a_j(\mathbf{x})dt$ to represent the probability that one reaction R_j will fire somewhere inside Ω in the infinitesimal time interval $[t, t + dt)$. In addition, a state change vector ν_j is defined to characterise reaction channel R_j . The element ν_{ij} of ν_j represents the change in the number of species S_i due to reaction R_j . The $N \times M$ matrix ν with elements ν_{ij} is called the stoichiometric matrix.

The SSA is a statistically exact procedure for generating the time and index of the next occurring reaction in accordance with the current values of the propensity functions. In the so-called direct method [4], we draw two independent random numbers r_1 and r_2 from the uniform distribution in the unit interval, and then take the time of the next reaction to be the current time plus μ , where

$$\mu = \frac{1}{a_0(\mathbf{x})} \ln \left(\frac{1}{r_1} \right),$$

and $a_0(\mathbf{x}) = \sum_{k=1}^M a_k(\mathbf{x})$. The index of the next reaction is the value of j that satisfies

$$\sum_{k=1}^{j-1} a_k(\mathbf{x}) < r_2 a_0(\mathbf{x}) \leq \sum_{k=1}^j a_k(\mathbf{x}).$$

Then the system is updated by $\mathbf{x}(t + \mu) = \mathbf{x}(t) + \nu_j$.

In the Poisson τ -leap method there are a number of reactions firing in a relatively larger time interval $[t, t + \tau)$. The reaction number of channel R_j

firing in $[t, t + \tau)$ is a sample value from a Poisson random variable $P(a_j(\mathbf{x})\tau)$ with mean $a_j(\mathbf{x})\tau$. After generating a sample value for each reaction channel, the system is updated by

$$\mathbf{x}(t + \tau) = \mathbf{x}(t) + \sum_{j=1}^M \nu_j P(a_j(\mathbf{x})\tau). \quad (1)$$

We leave the discussion for determining the leap size τ to the next section.

In the Poisson τ -leap method, state $\mathbf{x}(t)$ is used to approximate the states of the system in the time interval $[t, t + \tau)$. In order to improve the accuracy, a predicted state at a point in $[t, t + \tau)$ is used to approximate the states of the system. Similar to the midpoint Runge-Kutta method for solving ordinary differential equations, a predicted state at the midpoint $(t + \tau/2)$ is defined by

$$\bar{\mathbf{x}} = \mathbf{x} + \left\lfloor \frac{1}{2} \tau \sum_{j=1}^M a_j(\mathbf{x}) \nu_j \right\rfloor, \quad (2)$$

where $\lfloor x \rfloor$ is the largest integer in x . In the Poisson midpoint τ -leap method [5], sample values are generated from the Poisson random variable $P(a_j(\bar{\mathbf{x}})\tau)$ and the system is updated by

$$\mathbf{x}(t + \tau) = \mathbf{x}(t) + \sum_{j=1}^M \nu_j P(a_j(\bar{\mathbf{x}})\tau). \quad (3)$$

The Poisson τ -leap and midpoint τ -leap methods are special cases of the following s -stage Poisson Runge-Kutta methods [2], defined by

$$\begin{aligned} Y_i &= \mathbf{x}(t) + \sum_{k=1}^M \nu_k P \left(\sum_{j=1}^s w_{ij} a_k(Y_j) \tau \right), \quad i = 1, \dots, s, \\ \mathbf{x}(t + \tau) &= \mathbf{x}(t) + \sum_{k=1}^M \nu_k P \left(\sum_{j=1}^s \beta_j a_k(Y_j) \tau \right), \end{aligned} \quad (4)$$

where Y_i represent internal approximations to the solutions between t and $t + \tau$, w_{ij} and β_i are method parameters which can be chosen to achieve good accuracy and stability properties. In addition, the Heun and R2 methods have been studied [2].

When applying the Poisson τ -leap methods to stochastic chemical kinetics, negative molecular numbers may be obtained if certain species have small molecular numbers and the stepsize is not very small. The main reason for this is that the range of sample values of a Poisson random variable is from zero to infinity, but the numbers of molecules are finite. The reaction number of each channel, generated from Poisson random variables, should not exceed the numbers of molecules that are involved in this reaction channel. For tackling the problem of negative numbers of molecules, we have introduced binomial random variables to restrict the possible reaction numbers in the next time interval [12]. A binomial random variable $B(N, p)$ denotes N repeated independent Bernoulli trials and each trial has probability of success p . The finite range of sample values of binomial random variable allows us to properly bound the reaction numbers in order to avoid negative populations.

In the binomial τ -leap method [12], the reaction number of channel R_j is a binomial random variable $B(N_j, b_j(\mathbf{x})\tau)$ under the stepsize condition $0 \leq b_j(\mathbf{x})\tau \leq 1$. Tian and Burrage [12] also proposed a sampling technique for the simultaneous occurrence of different reaction channels if a reactant species undergoes two or more reaction channels. Then in the binomial τ -leap method, we choose a stepsize τ satisfying the Leap Condition and stepsize condition for each reaction channel; generate a sample value for each channel from the binomial random variable $B(N_j, b_j(\mathbf{x})\tau)$; apply the sampling technique for the reaction channels if there are reactant species undergoing two or more reaction channels; and finally update the system by

$$\mathbf{x}(t + \tau) = \mathbf{x}(t) + \sum_{j=1}^M \nu_j B(N_j, b_j(\mathbf{x})\tau). \quad (5)$$

In addition, the binomial midpoint τ -leap method has been proposed [12]. Numerical results for three test systems indicated that the binomial τ -leap method has very good accuracy and has achieved significant improvement of efficiency over existing approaches [12].

3 Stepsize selection procedures

The starting point of the stepsize selection procedures is the Leap Condition proposed by Gillespie [5]

$$\Delta a_j(\tau, \mathbf{x}) \triangleq |a_j(\mathbf{x} + \lambda) - a_j(\mathbf{x})| \leq \epsilon a_0(\mathbf{x}), \quad j = 1, \dots, M, \quad (6)$$

where ϵ is a prespecified error control parameter ($0 < \epsilon \ll 1$), and $\mathbf{x} + \lambda$ is a predicted state at $t + \tau$ computed by using the Poisson τ -leap method. The net change in state in $[t, t + \tau)$ is

$$\lambda_i = \sum_{k=1}^M \nu_{ik} P(a_k(\mathbf{x})\tau), \quad i = 1, \dots, N. \quad (7)$$

Based on the mean and standard variance of the random variable $\Delta a_j(\tau, \mathbf{x})$, Gillespie and Petzold [6] proposed an improved leap size selection procedure:

$$\tau = \min \{ \tau_1, \tau_2 \}, \quad (8)$$

where

$$\begin{aligned} \tau_1 &= \min_{j \in [1, M]} \left\{ \frac{\epsilon a_0(\mathbf{x})}{|\mu_j(\mathbf{x})|} \right\}, \\ \tau_2 &= \min_{j \in [1, M]} \left\{ \frac{\epsilon^2 a_0^2(\mathbf{x})}{\sigma_j^2(\mathbf{x})} \right\}, \\ f_{ji}(\mathbf{x}) &= \sum_{k=1}^N \frac{\partial a_j(\mathbf{x})}{\partial x_k} \nu_{ki}, \quad j, i = 1, \dots, M, \end{aligned}$$

$$\begin{aligned}\mu_j(\mathbf{x}) &= \sum_{k=1}^M f_{jk}(\mathbf{x})a_k(\mathbf{x}), & j = 1, \dots, M, \\ \sigma_j^2(\mathbf{x}) &= \sum_{k=1}^M f_{jk}^2(\mathbf{x})a_k(\mathbf{x}), & j = 1, \dots, M.\end{aligned}$$

By defining a derivative matrix $F = (F_{ji})_{M \times N} = \left(\frac{\partial a_j(\mathbf{x})}{\partial x_i} \right)_{M \times N}$, we have

$$f = (f_{ji})_{M \times M} = F \times \nu. \quad (9)$$

As all the three matrices F , f and ν are sparse, the computational time will decrease dramatically using sparse data structures. Thus one approach is to write a detailed program for a specific reaction system by considering the sparse structure of these three matrices, and an alternative approach is to design a general-purpose computer program by using sparse matrix multiplication techniques [1]. The first approach has been used in stochastic simulations although it is not recommended. Additional difficulties will arise with this approach if we modify the system by adding or deleting a few reactions. For the second approach, some care is needed when handling sparse data structures.

This work designs a simple stepsize selection procedure that is an approximation to (8). The main idea is to use the difference of the propensity functions to approximate their derivatives. We first consider the mean of the net change in state in $[t, t + \tau)$ (7):

$$\bar{\lambda}_i(\tau) = \tau \sum_{k=1}^M \nu_{ik} a_k(\mathbf{x}), \quad i = 1, \dots, N. \quad (10)$$

The first order Taylor expansion of the increment of the propensity function is

$$a_j(\mathbf{x} + \bar{\lambda}(\tau)) - a_j(\mathbf{x}) \approx \sum_{i=1}^N \frac{\partial a_j}{\partial x_i} \bar{\lambda}_i(\tau) = \tau \sum_{i=1}^N \frac{\partial a_j}{\partial x_i} \sum_{k=1}^M \nu_{ik} a_k(\mathbf{x}) = \tau \mu_j(\mathbf{x}).$$

For a prespecified stepsize τ_0 , we estimate $\mu_j(\mathbf{x})$ simply by

$$\bar{\mu}_j(\mathbf{x}) = \frac{a_j(\mathbf{x} + \bar{\lambda}(\tau_0)) - a_j(\mathbf{x})}{\tau_0}. \quad (11)$$

This procedure is simpler than the existing procedures that are based on the calculation of partial derivatives of the propensity functions.

As to the stepsize that is based on the variance, we first give the detailed formula of $\sigma_j^2(\mathbf{x})$ for the three types of elementary reactions.

1. The first order reaction: $S_i \xrightarrow{c_j} S$

$$a_j(\mathbf{x}) = c_j x_i, \quad \sigma_j^2(\mathbf{x}) = c_j^2 \sum_{l=1}^M \nu_{il}^2 a_l(\mathbf{x}) \tau,$$

where c_j is the reaction rate of the j th reaction channel R_j .

2. The second order reaction: $S_i + S_k \xrightarrow{c_j} S$

$$a_j(\mathbf{x}) = c_j x_i x_k, \quad \sigma_j^2(\mathbf{x}) = c_j^2 \sum_{l=1}^M (x_k \nu_{il} + x_i \nu_{kl})^2 a_l(\mathbf{x}) \tau.$$

3. The homodimer formation: $S_i + S_i \xrightarrow{c_j} S$

$$a_j(\mathbf{x}) = \frac{1}{2} c_j x_i (x_i - 1), \quad \sigma_j^2(\mathbf{x}) = c_j^2 \sum_{l=1}^M (x_i - \frac{1}{2})^2 \nu_{il}^2 a_l(\mathbf{x}) \tau.$$

In order to estimate the value of $\sigma_j^2(\mathbf{x})$, we consider the prediction

$$\hat{\lambda}_i(\tau_0) = \tau_0 \sum_{l=1}^M \nu_{il}^2 a_l(\mathbf{x}), \quad i = 1, \dots, N, \quad (12)$$

where τ_0 is a prespecified stepsize. Then $\sigma_j^2(\mathbf{x})$ is approximated by

$$\bar{\sigma}_j^2(\mathbf{x}) = \frac{c_j \bar{X}_j |a_j(\mathbf{x} + \widehat{\lambda}(\tau_0)) - a_j(\mathbf{x})|}{\tau_0}, \quad j = 1, \dots, M, \quad (13)$$

where

$$\bar{X}_j = \begin{cases} 1, & \text{a first order reaction,} \\ \max\{x_i, x_k\}, & \text{a second order reaction,} \\ x_i, & \text{a homodimer formation.} \end{cases} \quad (14)$$

Using the first order Taylor expansion of $a_j(\mathbf{x} + \widehat{\lambda}(\tau_0))$, the value of $\bar{\sigma}_j^2(\mathbf{x})$ is

$$\bar{\sigma}_j^2(\mathbf{x}) = \begin{cases} c_j^2 \sum_{l=1}^M \nu_{il}^2 a_l(\mathbf{x}) \tau, & \text{a first order reaction,} \\ c_j^2 \sum_{l=1}^M \bar{X}_j (x_k \nu_{il}^2 + x_i \nu_{kl}^2) a_l(\mathbf{x}) \tau, & \text{a second order reaction,} \\ c_j^2 \sum_{l=1}^M (x_i - \frac{1}{2}) x_i \nu_{il}^2 a_l(\mathbf{x}) \tau, & \text{a homodimer formation.} \end{cases}$$

Thus $\sigma_j^2(\mathbf{x})$ and $\bar{\sigma}_j^2(\mathbf{x})$ are equal for a first order reaction. For the homodimer formation, $\bar{\sigma}_j^2(\mathbf{x})$ is slightly larger than $\sigma_j^2(\mathbf{x})$, but these two values are different in the case of a second order reaction. If either ν_{il} or ν_{kl} is zero, the value of $\bar{\sigma}_j^2(\mathbf{x})$ is slightly larger than $\sigma_j^2(\mathbf{x})$, and since the stoichiometric matrix is sparse, the difference in the values of $\bar{\sigma}_j^2(\mathbf{x})$ and $\sigma_j^2(\mathbf{x})$ should be small. Numerical results in Section 4 indicates that the impact of this difference is negligible on the determination of the stepsizes.

Finally, we propose the following leap size selection procedure.

1. Based on the state \mathbf{x} at time t and a prespecified stepsize τ_0 , make the predictions by using (10) and (12).
2. Calculate the values of the propensity functions based on the predicted states $\mathbf{x} + \bar{\lambda}(\tau_0)$ and $\mathbf{x} + \widehat{\lambda}(\tau_0)$, and estimate $\mu_j(\mathbf{x})$ and $\sigma_j^2(\mathbf{x})$ by using (11) and (13), respectively.
3. Choose the leap size as

$$\tau = \min \{ \bar{\tau}_1, \bar{\tau}_2 \}, \quad (15)$$

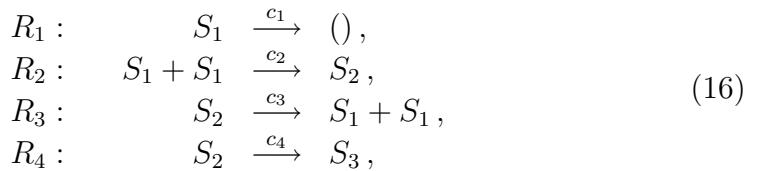
where

$$\bar{\tau}_1 = \min_{j \in [1, M]} \left\{ \frac{\epsilon a_0(\mathbf{x})}{|\bar{\mu}_j(\mathbf{x})|} \right\}, \quad \bar{\tau}_2 = \min_{j \in [1, M]} \left\{ \frac{\epsilon^2 a_0^2(\mathbf{x})}{\bar{\sigma}_j^2(\mathbf{x})} \right\}.$$

Sparse matrix techniques have been applied to the multiplication of the sparse stoichiometric matrix ν and the vector of propensity functions $a(\mathbf{x})$ in the above procedure and at the stage of updating the system (5). We used the old Yale sparse matrix format to represent the stoichiometric matrix using three vectors [1]. In the row format, all the non-zero elements are stored in a vector M . The column indices of M are stored in the $J\nu$ vector and a vector $I\nu$ is for the row indices of the stoichiometric matrix. The length of row i is determined by $I\nu(i+1) - I\nu(i)$. A similar approach is to use the column format for the multiplication of two matrices F and ν in (9).

4 Simulation results

The first test system contains three reactant species and four reaction channels, defined by



with reaction rates $\mathbf{c} = (0.1, 0.002, 0.5, 0.04)^\top$ and initial conditions $\mathbf{x}(0) = (10^4, 0, 0)^\top$.

This test system is used to compare the stepsizes obtained by the two stepsize selection procedures (8) and (15). The binomial τ -leap method was used to simulate this system with different prespecified error control parameters ϵ . At each step we use the improved selection procedure (8) to determine

the leap size for simulation. At the same time, we calculate the leap size by using the proposed selection procedure (15) and measure the difference of stepsizes by $D(\tau_1) = \tau_1 - \bar{\tau}_1$, $D(\tau_2) = \tau_2 - \bar{\tau}_2$ and $D(\tau) = \tau - \bar{\tau}$.

Figure 1 gives the stepsizes and the difference of stepsizes with three prespecified error control parameter $\epsilon = 0.05$ (A and D), 0.03 (B and E) and 0.01 (C and F), respectively. For each value of the error control parameter ϵ , $\bar{\tau}_1$ always gives a very good approximation to τ_1 . On the other hand, τ_2 is always larger than $\bar{\tau}_2$. The reason for this is that τ_2 is mainly determined by the homodimer formation in this system, and the value of $\bar{\sigma}_j^2$ is greater than that of σ_j^2 for this reaction. Our simulation experience indicates that $\epsilon = 0.03$ or preferably $\epsilon = 0.01$ should be used in the binomial τ -leap method in order to achieve good accuracy. For these two values of ϵ , $\bar{\tau}_2$ gives a very good approximation to τ_2 . In addition, there are large variations in the values of τ_1 but the values of τ_2 are relatively stable, which is consistent with observations [6]. Furthermore, results in Figures 1 and 2 indicate that the difference of stepsize $D(\tau)$ is between $D(\tau_1)$ and $D(\tau_2)$ and is mainly equal to $D(\tau_2)$ when $\epsilon = 0.01$.

We also tested the influence of the prespecified stepsize τ_0 on the accuracy of the estimated stepsizes $\bar{\tau}_1$ and $\bar{\tau}_2$. A large prespecified stepsize τ_0 causes relative large errors in the estimated stepsizes. However, a small prespecified stepsize τ_0 does not guarantee good estimation because of round-off errors in the computation. In this work we choose $\tau_0 = 0.01$ in all simulations.

The next system describes the expression of LacZ and LacY genes and activity of LacZ and LacY proteins in *E. coli*. A detailed description of this system [8, 12] has 22 reactions and 23 chemical species. As the SSA is frequently used at the initial stage of each simulation, we simulated this system in the time interval [300, 800], and use the SSA to obtain a state of the system at $t = 300$ that is used as the initial conditions of our simulations.

This system is used to compare the efficiency of the two procedures for determining stepsizes. As discussed in Section 3, sparse matrix techniques

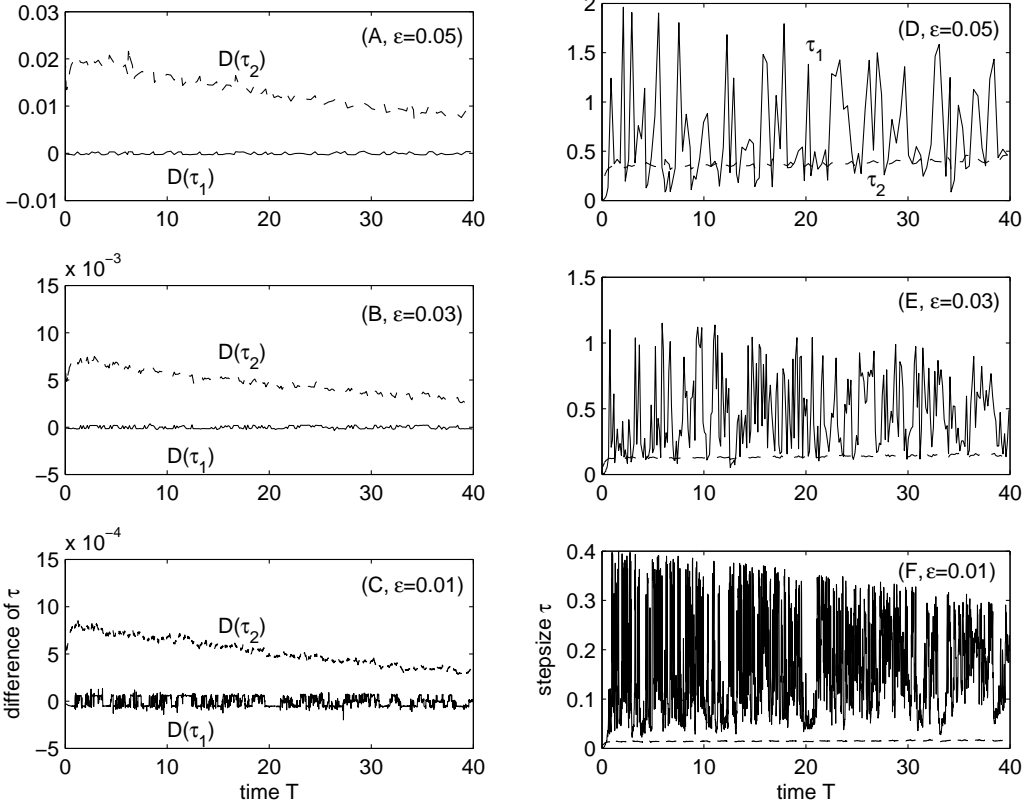


FIGURE 1: The difference of stepsizes $D(\tau_1)$ (solid line) and $D(\tau_2)$ (dash-line) in A, B and C, and the values of stepsizes τ_1 (solid line) and τ_2 (dash-line) in D, E and F.

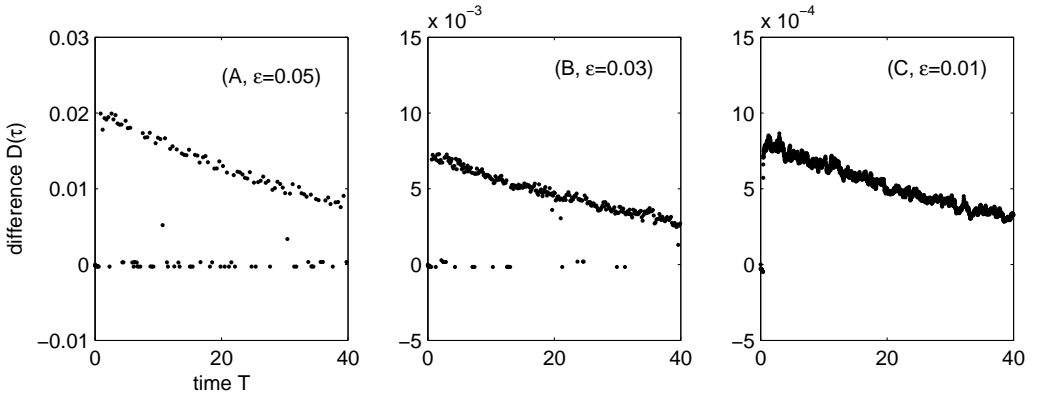


FIGURE 2: The difference of stepsizes $D(\tau)$ for $\epsilon = 0.05$ (A), $\epsilon = 0.03$ (B) and $\epsilon = 0.01$ (C).

were used in the new procedure. We also considered the application of the sparse matrix techniques to the improved selection process (8). The following three schemes were employed in all the simulations.

1. Scheme 1 is based on the improved procedure (8) by using the sparse structure of the stoichiometric matrix ν in the computation of f_{ji} and in updating system (5).
2. Scheme 2 is also based on the improved procedure (8). Sparse matrix techniques are used in the computation of f_{ji} , μ_j and σ_j^2 , and in updating the system.
3. Scheme 3 is based on the new procedure (15). The sparse structure of the stoichiometric matrix is considered in the prediction steps (10) and (12), and in updating the system.

For schemes 1 and 2, we implemented a subroutine based on the analytic derivatives of propensity functions for simplicity. Instead of writing a

complicated program using the sparse matrix multiplication package [1], we used a carefully designed subroutine based on the ideas in that article for the computation of f_{ji} , μ_j and σ_j^2 by using the sparse structure of the three matrices f , F and ν . Thus Scheme 2 is an ideal program for achieving maximal computational efficiency. Schemes 1 and 2 are at opposite ends of the efficiency spectrum, and any scheme that employs sparse matrix techniques as part of the computation [12, e.g.], will achieve an efficiency that is between those of schemes 1 and 2.

Table 1 gives the mean computational time of these three schemes over 100 simulations. Scheme 1 takes about four times the computing time of schemes 2 and 3. These times indicate that the existing leap size selection procedure in an improper implementation may take the majority of the total computing time, and the development of the stepsize selection procedure is an important issue in improving the efficiency of the τ -leap methods. The computing time of Scheme 3 is slightly larger than that of Scheme 2. The newly developed procedure can approach the maximal efficiency of the existing procedure and is easy to implement. Note that this maximal efficiency of the existing process is achieved by a handwritten program that is based on sparse matrix techniques and analytic derivatives of the propensity functions. Similar to the results for the first test system, the averaged numbers of steps in one simulation are very close to each other for these two leap size selection procedures. The difference in the averaged numbers of steps is less than 0.1% of the total number of steps. The numbers of steps of Scheme 2 are the same as those of Scheme 1 and thus are not listed in Table 1.

5 Conclusion

We studied procedures for determining stepsizes in the τ -leap methods. The main idea is to use the difference of propensity functions to approximate their derivatives. With two properly predicted states of the system, the

TABLE 1: Computing time (in seconds) and averaged numbers of steps for one simulation of the second system.

	Computing time			Number of steps	
	Scheme 1	Scheme 2	Scheme 3	Scheme 1	Scheme 3
$\epsilon = 0.05$	95.50	24.55	24.78	302970	302685
$\epsilon = 0.04$	99.68	25.59	25.77	316434	316351
$\epsilon = 0.03$	122.80	31.11	31.49	390825	391752
$\epsilon = 0.02$	238.56	60.33	60.30	762169	765788
$\epsilon = 0.01$	949.46	235.67	237.73	3055686	3051056

stepsizes obtained by the proposed procedure give very good approximations to those obtained by current procedures. The new procedure can approach the maximal efficiency achieved by a handwritten program based on sparse matrix techniques and analytic derivatives of the propensity functions. More importantly, this newly developed procedure is easy to implement in a general purpose computer program. Furthermore, we expect that as the chemical systems become larger, the savings in computational time observed for the *E. coli* model will become even more pronounced. Future work will focus on the development of sampling techniques in the binomial τ -leap method for simulating large-scale biochemical reaction systems.

Acknowledgements K. B. acknowledges funding from the Australian Research Council under the Federation Fellowship Program.

References

- [1] R. E. Bank and C. C. Douglas, Sparse matrix multiplication package (SMMP), *Adv. Comput. Math.* **1** (1993) 127–137.

- [doi:10.1007/BF02070824](https://doi.org/10.1007/BF02070824) C1029, C1032, C1036
- [2] K. Burrage and T. Tian, Poisson Runge-Kutta methods for chemical reaction systems, in *Advances in Scientific Computing and Applications*, Y. Lu et al., eds, Science Press, Beijing/New York, 82–96, 2004. C1024, C1026, C1027
- [3] K. Burrage, T. Tian and P. Burrage, A Multi-scaled approach for simulating chemical reaction systems, *Prog. Biophys. Mol. Bio.* **85** (2004) 217–234. [doi:10.1016/j.pbiomolbio.2004.01.014](https://doi.org/10.1016/j.pbiomolbio.2004.01.014) C1024
- [4] D. T. Gillespie, Exact stochastic simulation of coupled chemical reactions, *J. Phys. Chem.* **81** (1977) 2340–2361. [doi:10.1021/j100540a008](https://doi.org/10.1021/j100540a008) C1023, C1025
- [5] D. T. Gillespie, Approximate accelerated stochastic simulation of chemical reaction systems, *J. Chem. Phys.* **115** (2001) 1716–1733. [doi:10.1063/1.1378322](https://doi.org/10.1063/1.1378322) C1023, C1024, C1026, C1028
- [6] D. T. Gillespie and L. R. Petzold, Improved leap-size selection for accelerated stochastic simulation, *J. Chem. Phys.* **119** (2003) 8229–8234. [doi:10.1063/1.1613254](https://doi.org/10.1063/1.1613254) C1024, C1028, C1033
- [7] E. L. Haseltine and J. B. Rawlings, Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics, *J. Chem. Phys.* **117** (2002) 6959–6969. [doi:10.1063/1.1505860](https://doi.org/10.1063/1.1505860) C1024
- [8] A. M. Kierzek, STOCKS: stochastic kinetic simulations of biochemical systems with Gillespie algorithm, *Bioinformatics* **18** (2002) 470–481. C1033
- [9] J. Puchalka and A. M. Kierzek, Bridging the gap between stochastic and deterministic regimes in the kinetic simulations of the biochemical reaction networks, *Biophys. J.* **86** (2004) 1357–1372. C1024

- [10] C. Rao and A. Arkin, Stochastic chemical kinetics and the quasi-steady-state assumption: application to the Gillespie algorithm, *J. Chem. Phys.* **118** (2003) 4999–5010. doi:10.1063/1.1545446 C1024
- [11] M. Rathinam, L. R. Petzold, Y. Cao and D. T. Gillespie, Stiffness in stochastic chemically reacting systems: The implicit tau-leaping method, *J. Chem. Phys.* **119** (2003) 12784–12794. doi:10.1063/1.1627296 C1024
- [12] T. Tian and K. Burrage, Binomial leap methods for simulating chemical kinetics, *J. Chem. Phys.* **121** (2004) 10356–10364. doi:10.1063/1.1810475 C1024, C1027, C1028, C1033, C1036

Author addresses

1. **Tianhai Tian**, Advanced Computational Modelling Centre, The University of Queensland, AUSTRALIA.
<mailto:tian@maths.uq.edu.au>
2. **Kevin Burrage**, Advanced Computational Modelling Centre, The University of Queensland, AUSTRALIA.
<mailto:kb@maths.uq.edu.au>