# Hybrid Moment Computation Algorithm for Biochemical Reaction Networks

Yun-Bo Zhao, Jongrae Kim and João P. Hespanha

***Abstract*—Moment computation is essential to the analysis of stochastic kinetic models of biochemical reaction networks. It is often the case that the moment evolution, usually the first and the second moment evolutions over time, is all the information of interest. However, potential approaches to moment computation, specifically, the moment closure method and the exact stochastic simulation method, have their significant deficiency. The former, despite its computational efficiency, is essentially an approximation to the real solution and thus is lack of inaccuracy at certain conditions, while the computational inefficiency makes the usage of the latter limited to the networks with small number of molecules. A hybrid moment computation algorithm is therefore proposed by integrating the moment closure method and the exact stochastic simulation algorithms. The moment closure method and the stochastic simulation algorithm operate by turns to achieve an optimal balance between the efficiency due to the moment closure method and the accuracy due to the stochastic simulation. The hybrid algorithm is applied to a *Dictyostelium* cAMP oscillation network. The simulation results illustrate the effectiveness of the algorithm.**

## I. Introduction

The time evolution of biochemical reaction networks (BRNs) can be either described by continuous and deterministic differential equations or discrete and stochastic ones. The latter are more favoured in the sense that they represent the physical reality more accurately [1], [2]. However, although the chemical master equation (CME), which is used to describe the biochemical reactions in the discrete stochastic framework, is precise and simple, it is often analytically intractable even for fairly simple BRNs. Due to this reality, stochastic simulation algorithms (SSAs) are preferred instead in practise. In addition, for many cases the SSA is the only possible way to investigate the characteristics of BRNs [2].

Many exact SSAs have been developed to date and they are exact in the sense that they simulate exact independent realisations of the underlying stochastic kinetic model [1], [3]. However, the SSA can be computationally expensive for those BRNs with a large number of molecules, which are often seen in practise. Therefore, their approximation counterparts, i.e., approximated SSAs, have been also developed in order to deal with the efficiency issue in the exact SSAs [4], [5]. Each run of the SSA can produce a realisation of the underlying stochastic kinetic model for the concerned BRN, and thus provide us the information, in the stochastic sense, of how the BRN evolves over time. By running as many realisations as possible, the vivid information of the probability distribution of the BRN can then be obtained numerically. Unfortunately, this computing process, even for approximated SSAs, can be extremely time-consuming or even computationally impossible for some cases. On the other hand, what is really interested in is often only the first several moments (usually the first two, that is, the mean and the variance) of the species involved in the BRNs rather than the full probability distribution. However, to obtain these moments, using the SSA it is still inevitable to first run as many realisations as possible and then calculate the moments using statistical methods. This computational inefficiency thus makes the SSA not the best way to conduct the moment computation.

To deal with this issue, one may turn to approximated analytic solutions to the CME, thus resulting in various so-called moment closure (MC) methods [6]. These methods intend to approximate the moment evolution by ordinary differential equations (ODEs). This approach is efficient with the help of numerical computing methods for ODEs, and also gives rise to satisfactory results for a large number of BRNs. However, this approach is essentially an approximation to the real solution and therefore, it could fail to capture the moment evolution at certain conditions, due to the accumulation of the approximation errors over time. Hence, the main dilemma is as follows: the SSA guarantees accurate moment computation but it is time-consuming or even computationally impossible to obtain just one realisation in a reasonable length of time for some BRNs; on the other hand, the MC method is fast to compute but may not be able to capture some important behaviour in BRNs.

A hybrid moment computation (HMC) approach is therefore proposed in order to obtain the balance between the SSA and the MC methods. The underlying idea of this approach is to "rebuild" the accuracy of the MC method by running occasionally exact SSAs for the considered BRN. Using this approach, the MC method is first used to obtain the moment evolution until when it diverges from a predetermined threshold. A certain number of the exact SSA runs for a given time length and the stochastic information from the set of realisations is then used to reset the initial condition for the MC method. These two steps keep running till the simulation reaches to the final time. Intuitively, by designing the associated parameters appropriately the HMC approach can achieve a balance between the efficiency of the MC method and the accuracy of the exact SSAs.

The cAMP oscillation network observed in homogeneous populations of *Dictyostelium* cells 4h after the initiation of development is considered. An improvement in the balance of

Yun-Bo Zhao and Jongrae Kim are with the Department of Biomedical Engineering, University of Glasgow, Glasgow, G12 8QQ, U.K. {yzhao,jkim}@eng.gla.ac.uk

João Pedro Hespanha is with the Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93101, U.S.A. hespanha@ece.ucsb.edu

the efficiency and the accuracy of the HMC approach over both the SSA and the MC method is demonstrated in the simulation.

## II. Preliminaries

In this section, the SSA and the MC method for the stochastic kinetic model of BRNs are summarised. For simplicity of notations, in what follows a BRN is considered which involves $u$ species, i.e., $X_1$, $X_2$, …, $X_u$, $v$ reactions, i.e., $R_1$, $R_2$, …, $R_v$ and the molecule counts of the species are denoted by a column vector $x = [x_1, x_2, \ldots, x_u]^T$, where $x_i$ is the molecule count of $X_i$.

### A. SSAs for the stochastic kinetic model of BRNs

All the SSAs are based on the assumption that the molecular system is kept "well-mixed". Under such an assumption, one may define the "stochastic rate constant" $c_i$ for reaction $R_i$, and $c_i dt$ can then be interpreted as "the average probability that reaction $R_i$ occurs in the next infinitesimal time interval $dt$" [1]. Associated with $c_i$ is the reaction rate law, $h_i(x, c_i)$ which indicates that, conditional on the current state $x$ at time $t$, the probability that an $R_i$ reaction will occur in the time interval $(t, t + dt]$ is given by $h_i(x, c_i)dt$. It is thus clear that in the absence of any other reactions taking place, the time to the occurrence of the next $R_i$ reaction can be determined by an random variable satisfying the exponential distribution with parameter $h_i(x, c_i)$, denoted by $\text{Exp}[h_i(x, c_i)]$.

In order to perform the stochastic simulation, it is only necessary to answer the following two questions: 1) What is the time of the next reaction? and 2) Which is the next reaction? With the above assumptions and the properties of the exponential distribution, the following result can be readily concluded, where $\tau$ is the time of the next reaction and $j$ is the index of the next reaction:

$$\tau \sim \text{Exp}[h_0(x, c)] \tag{1a}$$

$$Pr(j = i) = \frac{h_i(x, c_i)}{h_0(x, c)} \tag{1b}$$

where $h_0(x, c) \triangleq \sum_{i=1}^{v} h_i(x, c_i)$. That is, $\tau$ is a random variable satisfying the exponential distribution with parameter $h_0(x, c)$ and $j$ is a discrete random variable with the probability mass function being defined as in (1b).

On the basis of the above discussion, Gillespie's direct method can then be summarised as follows, where the simulation initial and final times are set to be $T_{start}$ and $T_{end}$, respectively.

*Algorithm 1 (Gellespie's direct method):*

S1. Initialisation at $t = T_{start}$. Specify the stochastic rate constants $c_i, i = 1, 2, \ldots, v$ and the initial molecule counts of the species $x_i, i = 1, 2, \ldots, u$.

S2. Calculate $h_i(x, c_i), i = 1, 2, \ldots, v$ and $h_0(x, c)$.

S3. Determine the time of the next event $t + \tau$ by (1a), and the index of the next reaction $j$ by (1b).

S4. Update $t = t + \tau$ and $x$ according to reaction $R_j$.

S5. Return to S2 if $t < T_{end}$; otherwise stop the simulation.

As mentioned earlier, other SSAs are also available, either exact or approximated ones. The above classic SSA is used but all the other SSAs can also be fitted in the HMC approach without any significant modifications.

### B. Moment closure for the stochastic kinetic model of BRNs

In many cases the moment evolution of $x$ is of primary interest and it is reasonable to try to solve the CME analytically. In order to do so, construct a vector $\mu^m$ containing all the moments of $x$ up to some order $m$, that is, $\mu^m$ contains all the moments of the form, $E[x_1^{m_1} x_2^{m_2} \ldots x_u^{m_u}]$, where $\sum_{j=1}^{u} m_j = m$. As was shown in [6], the evolution of $\mu^m$ is determined by an ODE of the following form

$$\dot{\mu}^m = A\mu^m + B\bar{\mu}^m, \mu^m \in \mathbb{R}^M, \bar{\mu}^m \in \mathbb{R}^{\bar{M}} \tag{2}$$

where $A$ and $B$ are appropriately defined matrices and $\bar{\mu}^m$ is a vector containing moments of order larger than $m$. The dimension $M$ is always larger than $m$ as there are many moments of each order. In general $M$ is of order $u^m$ [7].

The moment evolution in (2) is exact but usually not closed as the term $B\bar{\mu}^m$ always appears whenever at least one reaction has 2 or more reactants. In order to obtain a computationally feasible formulation, the MC method is therefore introduced, which approximates the exact moment evolution $\mu^m$ in (2) by the following approximated one, $\nu^m$,

$$\dot{\nu}^m = A\nu^m + B\varphi(\nu^m), \nu^m \in \mathbb{R}^M \tag{3}$$

where $\varphi(\nu^m)$ is a column vector that approximates the moments in $\bar{\mu}^m$.

Various methods have been proposed to construct the function $\varphi(\nu^m)$, in order that the solution to (3) can be as close to the solution to (2) as possible. Among all these methods of particular interest is the zero cumulants method which is a distribution-based method for moment closure. It finds the $k$th order moment closure function $\varphi(\nu^m)$ by assuming that all multi-variable cumulants of the population with order larger than $k$ are negligible [7]. This approach is believed to be superior in the case where the distributions have low variability, i.e., low standard deviations compared to the mean, and is used in the example presented in Section IV.

## III. Hybrid Moment Computation for BRNs

The moment evolution of BRNs is the primary information of interest for many cases. It is possible to obtain this using the two analytic methods for the stochastic kinetic model of BRNs summarised in the previous section. Although it is possible to obtain the moment evolution by running SSAs for the BRN as many times as possible and then calculating the moment evolution using statistical methods, this approach is usually time consuming and computationally inefficient, especially for the BRNs with relatively large number of molecules. The MC method, therefore, is a preferred alternative to this problem due to its efficiency. Unlike the SSAs, this approach only needs corresponding ODEs to be solved. However, the MC method is an approximation to the true solution and it is observed that at certain conditions, this approach fails to capture the true moment evolution after a certain time of development, most likely due to the approximation error accumulating over

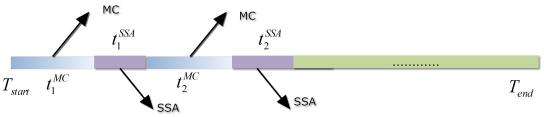


Fig. 1. The basic idea of the hybrid moment computation approach.

time (Fig. 3). Therefore, a more accurate and efficient moment computation approach is yet to be designed.

It is not difficult to see why a hybrid approach combining the SSA and the MC method is of particular interest in dealing with this issue. Due to the fact that the SSA simulates exact independent realisations of the underlying stochastic kinetic model, it can achieve a high accuracy of the moment computation. On the other hand, the MC method is very efficient as only a set of ODEs is involved. Therefore, it is straightforward that appropriate combination of these two methods can potentially provide both accuracy and efficiency. Thus, it could be a promising approach to the moment computation for BRNs. This combination is illustrated in Fig. 1, where the two methods operate by turns to achieve an optimal combination of the advantages of both methods. Denote the $i$th run of the MC method and the SSA by $MC_i$ and $SSA_i$, the starting instant of them by $T_i^{MC}$ and $T_i^{SSA}$, and the length of them by $t_i^{MC}$ and $t_i^{SSA}$, respectively. The computation then consists of a "run" sequence of the form $\{MC_1, SSA_1, MC_2, SSA_2, \ldots, MC_i, SSA_i, \ldots\}$ and the switches are made at time instants $T_i^{MC}$ (from $SSA_{i-1}$ to $MC_i$) and $T_i^{SSA}$ (from $MC_i$ to $SSA_i$), respectively. The total time of running each method can be obtained as $t^{MC} = \sum_i t_i^{MC}$ and $t^{SSA} = \sum_i t_i^{SSA}$. If we denote the simulation time by $t_{sim} \triangleq T_{end} - T_{start}$, it is then held that $t^{MC} + t^{SSA} = t_{sim}$. The objective of this hybrid approach would be, given the fixed simulation time $t_{sim}$, making the time of running the SSA, $t^{SSA}$, as little as possible, or equivalently, the time of running the MC method, $t^{MC}$, as much as possible, under the condition that the error of the moment evolution stays in a given tolerance range.

Despite the simplicity of the idea, the practical implementation of this approach is nothing straightforward. In fact, before this approach can be actually applied to the practise, two key issues have to be carefully resolved: firstly, for each method, how we can initiate the run at the beginning of $T_i^{MC}$ and $T_i^{SSA}$, and secondly, how long these time intervals $t_i^{MC}$ and $t_i^{SSA}$ should be? These issues are discussed in detail for the case that only the moments of the first two orders are of interest. The general case can be discussed similarly.

### A. Determination of $t_i^{MC}$ for $MC_i$

In view of the above discussion, a design principle of the HMC approach is to maximise the running time of the MC method, $t^{MC}$, and therefore it is reasonable to maximise $t_i^{MC}$ for each $MC_i$. However, running the MC method too long without corrections may cause the error of the moment evolution grows beyond the tolerance. Therefore, an optimal value of $t_i^{MC}$ is to be found.

Despite this demand, the reality is that there is no additional information available in the determination of the optimal value of $t_i^{MC}$. Instead, as an alternative and suboptimal method, in the example presented in the next section a constant value of $t_i^{MC}$ is used,

$$t_i^{MC} = t_c^{MC}, \forall i \tag{4}$$

But it is worth mentioning that the optimal value of $t_i^{MC}$ is still to be found in the future work.

### B. Initiation at $T_i^{SSA}$ for $SSA_i$

The introduction of the SSA to the HMC approach is to increase the accuracy in the moment computation as the SSA simulates exact independent realisations and is exact in the stochastic sense. It is noticed that this accuracy is partly due to the exact initial state information, the acquisition of which, however, turns to be a difficulty for the HMC approach.

In fact, the MC method is a deterministic integration that is able to provide the approximated moment evolution for BRNs. Therefore, even though a stochastic distribution based on this moment information can be generated at a specific instant, it could be different from the true one due to the essential randomness, the unknown type of the distribution in general and, furthermore, the inaccurate moment information. To deal with the randomness, it is thus necessary to run the SSA for as many times as possible (the number of which, denote by $n_{SSA}$, is certainly constrained by the efficiency requirement). As regards the type of the distribution, the multi-variable normal distribution is used for the generation of the initial distribution. Suppose the solution to (3), $\nu^2 \triangleq [\nu_1 \ \nu_2 \ \ldots \ \nu_{n_u}]$ is to approximate $\mu^2 \triangleq [EX_1 \ EX_2 \ \ldots \ EX_u \ EX_1^2 \ EX_1X_2 \ \ldots \ EX_{u-1}X_u \ EX_u^2]$, where $n_u \triangleq (u^2 + 3u)/2$ is the number of the moments of the first and second orders. The initial distribution for $SSA_i$ is thus defined as the following multi-variable normal distribution,

$$N(\nu_m^2, \nu_c^2) \tag{5}$$

where the mean and the covariance matrix, $\nu_m^2$ and $\nu_c^2$ can be obtained from $\nu^2$ directly.

In order to perform $SSA_i$, $n_{SSA}$ different initial states are generated from (5), which then produce $n_{SSA}$ realisations of the underlying stochastic kinetic model by Algorithm 1 during $SSA_i$, denoted by $\hat{x}_{SSA_i}^j(t)$, $j = 1, 2, \ldots, n_{SSA}, t \in [T_i^{SSA} \ T_{i+1}^{MC}]$.

### C. Determination of $t_i^{SSA}$ for $SSA_i$

The determination of $t_i^{SSA}$ is less complicated, although a balance is still there between the accuracy by increasing $t_i^{SSA}$ and the efficiency by decreasing $t_i^{SSA}$. A simple way of determining $t_i^{SSA}$ would be letting it be constant, that is,

$$t_i^{SSA} = t_c^{SSA}, \forall i \tag{6}$$

However the optimal value of $t_c^{SSA}$ is still to be found.

### D. Initiation at $T_i^{MC}$ for $MC_i$

As an ODE, the running of the MC at $T_i^{MC}$ requires only the moment information at $T_i^{MC}$ as its initial state. This information can be obtained from the $n_{SSA}$ realisations of the SSA in the last time interval $SSA_{i-1}$, which is a simple application of the standard statistical methods. For vector $x$, define $x \odot x = [x_1\ x_2\ \ldots\ x_u\ x_1^2\ x_1x_2\ \ldots\ x_1x_u\ x_2^2\ x_2x_3\ \ldots x_u^2]$. The moments at $T_i^{MC}$ for the case where only the moments of the first two orders are of interest, $\nu^2_{SSA}(T_i^{MC})$ can then be obtained as follows,

$$\nu^2_{SSA}(T_i^{MC}) = \sum_{j=1}^{n_{SSA}} \frac{\left[\hat{x}^j_{SSA_i}(T_i^{MC}) \odot\ \hat{x}^j_{SSA_i}(T_i^{MC})\right]}{n_{SSA}} \quad (7)$$

It is realized that, although $\nu^2_{SSA}(T_i^{MC})$ can be served as the initial state for $MC_i$ by itself, this information is totally dependent on the limited number of realisations of the considered BRN and thus the negative stochastic effects are inevitable. Therefore, a better way of using this information is to "correct" the initial state of the MC method but not completely "replace" it. In order to do so, an auxiliary run of the MC method is conducted during $SSA_{i-1}$ and the moments of the first two orders at $T_i^{MC}$ are recorded as $\nu^2_{MC}(T_i^{MC})$. The initial moment information for the MC run $MC_i$ is then obtained by using a Kalman filter like mechanism, as follows,

$$\begin{aligned}\nu^2(T_i^{MC}) &= \nu^2_{MC}(T_i^{MC}) \\ &\quad + K\left[\nu^2_{SSA}(T_i^{MC}) - \nu^2_{MC}(T_i^{MC})\right]\end{aligned} \quad (8)$$

where $K$ is the update gain to be determined.

The algorithm of the HMC approach for the case that only the moments of the first two orders are of interest is summarised as follows, whereas the general case follows the same process only that the discussions in the above subsections are necessary to be reconsidered.

*Algorithm 2 (Hybrid moment computation):*

S1. At $t = T_{start}$ initialise the initial state $x_0$, parameters $n_{SSA}$, $t_c^{SSA}$, $t_c^{MC}$, $K$ and let $i = 1$.

S2. At $T_i^{MC}$, if $i = 1$ then initialise the initial state for the MC using $x_0$; otherwise obtain the initial state by (8). Run the MC for $MC_i$. Run the MC for $SSA_i$ and record its final state $\nu^2_{MC}(T_i^{MC})$ for future use.

S3. At $T_i^{SSA}$, if $T_i^{SSA} \geq T_{end}$ then stop the simulation; otherwise initialise by (5), run the SSA for $SSA_i$ and let $i = i + 1$.

S4. If $T_i^{MC} \geq T_{end}$, stop the simulation; otherwise return to S2.

## IV. Example

In this section, a *Dictyostelium* cAMP oscillation network is considered, as a typical example to verify the effectiveness of the proposed approach.

### A. Dictyostelium cAMP oscillating network

In [8], Laub and Loomis propose a model of the molecular network underlying adenosine 3′, 5′-cyclic monophosphate (cAMP) oscillations observed in fields of chemotactic *Dictyostelium discoideum* cells. This model induces the spontaneous oscillations in cAMP observed 4h after the initiation of development. Based on the illustrative diagram depicted in Fig. 2, the deterministic description for the model is obtained in [8], as follows

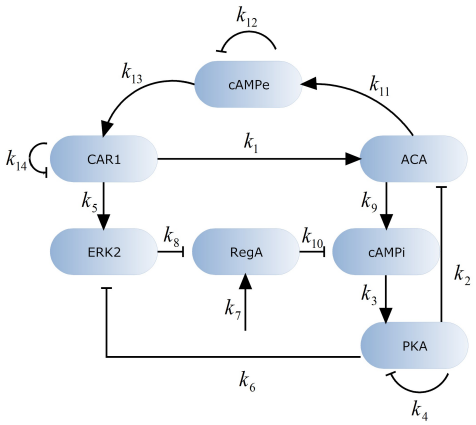


Fig. 2. Laub-Loomis model for the Dictyostelium cAMP oscillation network, where $\rightarrow$ and $\dashv$ represent positive and negative regulations respectively.

$$\begin{aligned}
d[\text{ACA}]/dt &= k_1[\text{CAR1}] - k_2[\text{ACA}][\text{PKA}] \\
d[\text{PKA}]/dt &= k_3[\text{cAMPi}] - k_4[\text{PKA}] \\
d[\text{ERK2}]/dt &= k_5[\text{CAR1}] - k_6[\text{PKA}][\text{ERK2}] \\
d[\text{RegA}]/dt &= k_7 - k_8[\text{ERK2}][\text{RegA}] \\
d[\text{cAMPi}]/dt &= k_9[\text{ACA}] - k_{10}[\text{RegA}][\text{cAMPi}] \\
d[\text{cAMPe}]/dt &= k_{11}[\text{ACA}] - k_{12}[\text{cAMPe}] \\
d[\text{CAR1}]/dt &= k_{13}[\text{cAMPe}] - k_{14}[\text{CAR1}]
\end{aligned}$$

where ACA is adenylyl cyclase, PKA is the protein kinase, ERK2 is the mitogen activated protein kinase, RegA is the cAMP phosphodiesterase, cAMPi and cAMPe are the internal and external cAMP concentrations, respectively, and CAR1 is the ligand cell receptor complex. The nominal values of the kinetic parameters $k_i, i = 1, 2, \ldots, 14$ are listed in Table IV-A. Note that these parameters are taken from [9].

TABLE I
KINETIC PARAMETERS: NOMINAL VALUES

| Parameter | Units | Nominal Value |
|---|---|---|
| $k_1$ | $\text{min}^{-1}$ | 2.0 |
| $k_2$ | $\mu\text{M}^{-1}\text{min}^{-1}$ | 0.9 |
| $k_3$ | $\text{min}^{-1}$ | 2.5 |
| $k_4$ | $\text{min}^{-1}$ | 1.5 |
| $k_5$ | $\text{min}^{-1}$ | 0.6 |
| $k_6$ | $\mu\text{M}^{-1}\text{min}^{-1}$ | 0.8 |
| $k_7$ | $\mu\text{M}^{-1}\text{min}^{-1}$ | 1.0 |
| $k_8$ | $\mu\text{M}^{-1}\text{min}^{-1}$ | 1.3 |
| $k_9$ | $\text{min}^{-1}$ | 0.3 |
| $k_{10}$ | $\mu\text{M}^{-1}\text{min}^{-1}$ | 0.8 |
| $k_{11}$ | $\text{min}^{-1}$ | 0.7 |
| $k_{12}$ | $\text{min}^{-1}$ | 4.9 |
| $k_{13}$ | $\text{min}^{-1}$ | 23.0 |
| $k_{14}$ | $\text{min}^{-1}$ | 4.5 |

The above deterministic model can be equivalently transformed to its corresponding stochastic model [2], [10], as follows

$$\begin{aligned}
\text{CAR1} &\xrightarrow{c_1} \text{ACA} + \text{CAR1}\\
\text{ACA} + \text{PKA} &\xrightarrow{c_2} \text{PKA}\\
\text{cAMPi} &\xrightarrow{c_3} \text{PKA} + \text{cAMPi}\\
\text{PKA} &\xrightarrow{c_4} \oslash\\
\text{CAR1} &\xrightarrow{c_5} \text{ERK2} + \text{CAR1}\\
\text{PKA} + \text{ERK2} &\xrightarrow{c_6} \text{PKA}\\
\oslash &\xrightarrow{c_7} \text{RegA}\\
\text{ERK2} + \text{RegA} &\xrightarrow{c_8} \text{ERK2}\\
\text{ACA} &\xrightarrow{c_9} \text{cAMPi} + \text{ACA}\\
\text{RegA} + \text{cAMPi} &\xrightarrow{c_{10}} \text{RegA}\\
\text{ACA} &\xrightarrow{c_{11}} \text{cAMPe} + \text{ACA}\\
\text{cAMPe} &\xrightarrow{c_{12}} \oslash\\
\text{cAMPe} &\xrightarrow{c_{13}} \text{CAR}_1 + \text{cAMPe}\\
\text{CAR1} &\xrightarrow{c_{14}} \oslash
\end{aligned}$$

where $\oslash$ represents relatively abundant source of molecules or a non-interacting product, $n_A = 6.023 \times 10^{23}$ is Avogadro's number, $V = 0.565 \times 10^{-12}$ $l$ is the average volume of a *Dictyostelium* cell [11], and $c_i, i = 1, 2, \ldots, 14$ are the stochastic rate constants for the stochastic model. According to different reaction orders (which are determined by the number of the species of the reactants involved in the reaction), the calculation of $c_i$ and $h(x, c_i)$ is obtained in Table II, where $i$ represents the index of the reactions (if we label the above reactions sequentially) and $j, k$ represent the reactants involved (the reactant in first-order reactions is $x_j$ and those in second-order reactions are $x_j$ and $x_k$).

TABLE II
PARAMETERS FOR THE STOCHASTIC KINETIC MODEL

| | $i$ | $c_i$ | $h(x, c_i)$ |
|---|---|---|---|
| Zero | 7 | $k_i n_A V 10^{-6}$ | $c_i$ |
| First | 1, 3, 4, 5, 9, 11, 12, 13, 14 | $k_i$ | $c_i x_j$ |
| Second | 2, 6, 8, 10 | $k_i/(n_A V 10^{-6})$ | $c_i x_j x_k$ |

### B. The MC method and the SSA

In this paper, the interest in the above model is to compute its first and second moment (consequently, the mean and the variance) evolution over time. As mentioned earlier, this can be done using either the SSA or the MC method. For the former, the Gillespie's direct method in Algorithm 1 is used while for the latter, the zero cumulants method is favoured as the standard deviations of the species are quite low compared to the means.

Let $T_{start}$ and $T_{end}$ be 0 and 230 minutes, respectively, and the initial molecule counts of the species be $x_0^{SSA}$ =[#ACA #PKA #ERK2 #RegA #cAMPi #cAMPe #CAR1]= [7290 7100 2500 3000 4110 1100 5960]. The SSA can then be performed by Algorithm 1, based on the stochastic kinetic model in the previous subsection and the initial state $x_0^{SSA}$. The "Dizzy" software is used to perform the SSA.

For the MC method with the stochastic kinetic model in the previous section, the stochastic rate constants and functions defined in Table II and the initial state defined as $\nu_0^{MC} = x_0^{SSA} \odot x_0^{SSA}$, it is then able to develop the moment closure equations as shown in (3). For brevity, these equations are omitted but the reader of interest is advised to construct these equations by themselves. These moment closure dynamics can also be automatically obtained by using the efficient MATLAB® toolbox `StochDynTools` [7].

The evolution of the mean and the variance of the considered model is illustrated in Fig. 3, using both the SSA and the MC method. It is seen that the mean evolution obtained using the MC method fits well with the one using the SSA but the variances obtained are not. Although the difference between the two variance evolutions is not so significant, it is noticed that, for oscillating BRNs, to a certain extent the variance is a more important index that indicates the oscillations in the network. This justifies the necessity of developing more accurate moment computation method for such BRNS, as the HMC method proposed in this paper.

It is worth pointing out that the moment evolution using the SSA is obtained by running 1000 times of the SSA in Algorithm 1 on a Dell Precision Workstation T3500 PC, configured with Windows Vista Business 64bit, Intel(R) Xeon(R) CPU W3503@2.4GHz, and 4GB RAM. Although 1000 times of realisations guarantees a pretty accurate time evolution of the model, the computation takes more than 2 days on such a fairly powerful PC and for such a relatively simple BRN, which clearly indicates the inefficiency of the SSA for the moment computation.

### C. The HMC approach

Before performing the HMC approach, 10 realizations of the considered network are obtained by running the Gillespie's algorithm. $K \triangleq [K_1 \ K_2] = [0 \ 0.15]$ is then determined using the trial-and-error method based on these realizations, where $K_1$ and $K_2$ are working on the first and second order of the moments, respectively. That is, for the initial condition for the MC method, the mean is unchanged while the covariance matrix is updated by (8). Other associated parameters are set as follows: $n_{SSA} = 100$, $t_c^{SSA} = 3$ minutes and $t_c^{MC} = 17$ minutes. As $n_{SSA}$ is relatively small, the above gain is chosen in a conservative way and the estimation relys more on the MC method. The evolution of the mean and the variance can now be obtained by Algorithm 2 and the results are illustrated in Fig. 4.

It is seen from Fig. 4 that the mean evolution still fits well with the one obtained using the SSA. Although the variance evolution does not precisely match the one obtained using the SSA, it has a significant improvement over the one obtained using the MC method in Fig. 3, and is acceptable to a certain extent in practice. Indeed, from the improved variance

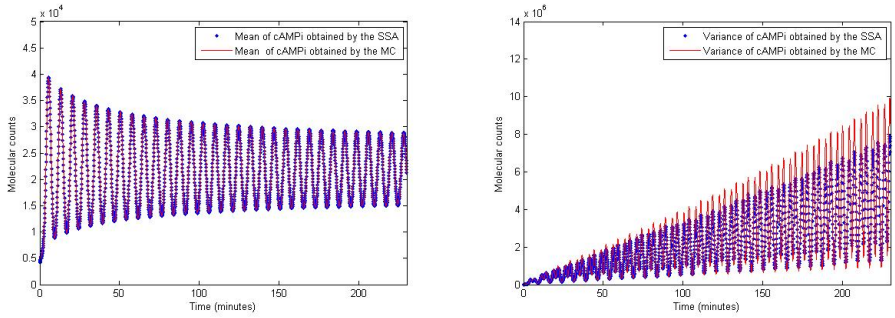


Fig. 3. The mean and the variance of cAMPi in the *Dictyostelium* cAMP oscillating network, obtained using the SSA and the MC method, respectively.

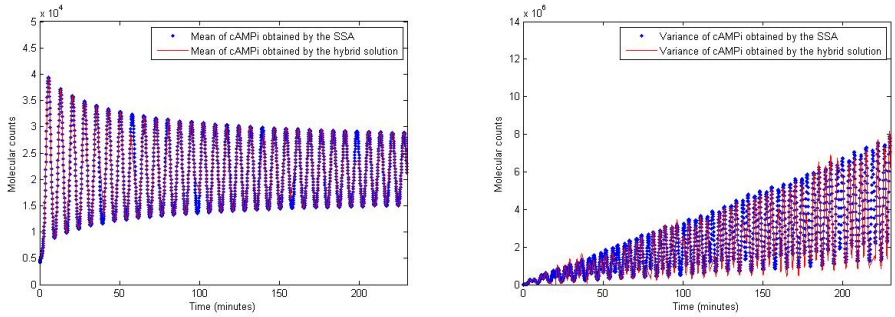


Fig. 4. The mean and the variance of cAMPi in the *Dictyostelium* cAMP oscillating network, obtained using the SSA and the HMC method, respectively.

evolution it is seen that the systematic error of the MC method in Fig. 3, that is, the difference of the variance obtained using the SSA and the MC method is increasing with time, has been effectively eliminated. This shows the effectiveness of the HMC approach to this oscillating BRN.

On the other hand, it is necessary to notice that the computation of the results in Fig. 4 takes less than 1.5 hour, which is a dramatically improvement in efficiency compared with more than 2 days using the SSA. Two obvious reasons contribute to this improvement, that is, the number of the realisations is significantly decreased using the HMC approach (as little as 100 times) and the time of running the SSA has been also dramatically decreased.

## V. Conclusions and Future Works

A hybrid moment computation approach is proposed for BRNs. This approach integrates the advantages of both the SSA and the MC method for moment computation and thus is potentially able to achieve the balance between the efficiency and the accuracy. As a preliminary result, a *Dictyostelium* cAMP oscillating network is considered which illustrates the effectiveness of the proposed approach.

Several important theoretical issues are to be considered in the future works, that is, firstly, the source of the inaccuracy brought by the MC method, secondly, the design of the optimal parameters for the HMC approach and, last but not least, the theoretical evaluation of the proposed HMC approach.

## Acknowledgements

The work of Yun-Bo Zhao and Jongrae Kim was supported by EPSRC research grant EP/G036195/1. João P. Hespanha's contribution was partially supported by the NSF grants ECCS-0725485 and ECCS-0835847.

## References

[1] D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *J. Phys. Chem.*, vol. 81, no. 25, pp. 2340–2361, 1977.

[2] D. J. Wilkinson, *Stochastic Modelling for Systems Biology*, 1st ed., ser. Mathematical and Computational Biology. Chapman & Hall, 2006.

[3] M. A. Gibson and J. Bruck, "Efficient exact stochastic simulation of chemical systems with many species and many channels," *J. Phys. Chem. A*, vol. 104, no. 9, pp. 1876–1889, 2000.

[4] D. T. Gillespie, "The chemical langevin equation," *J. Phys. Chem.*, vol. 113, no. 1, pp. 297–306, 2000.

[5] ——, "Approximate accelerated stochastic simulation of chemically reacting systems," *J. Phys. Chem.*, vol. 115, no. 4, pp. 1716–1733, 2001.

[6] A. Singh and J. P. Hespanha, "Lognormal moment closures for biochemical reactions," in *Proc. 45th IEEE Conf. Decision & Control*, San Diego, CA, USA, Dec. 13-15 2006, pp. 2063–2068.

[7] J. P. Hespanha, "`StochDynTools` — a MATLAB toolbox to compute moment dynamics for stochastic networks of bio-chemical reactions," Available at http://www.ece.ucsb.edu/ hespanha, May 2007.

[8] M. T. Laub and W. F. Loomis, "A molecular network that produces spontaneous oscillations in excitable cells of *Dictyostelium*," *Mol. Biol. Cell*, vol. 9, no. 12, pp. 3521–3532, 1998.

[9] L. Ma and P. A. Iglesias, "Quantifying robustness of biochemical network models," *BMC Bioinf.*, vol. 3, no. 38, 2002.

[10] J. Kim, D. G. Bates, and I. Postlethwaite, "Evaluation of stochastic effects on biomolecular networks using the generalised nyquist stability criterion," *IEEE Trans. Autom. Control*, vol. 53, no. 8, pp. 1937–1941, 2008.

[11] D. R. Soll, J. Yarger, and M. Mirick, "Stationary phase and the cell cycle of *Dictyostelium discoideum* in liquid nutrient medium," *J. Cell Sci.*, vol. 20, no. 3, pp. 513–523, 1976.