A Constrained Approach to Multiscale Stochastic Simulation of Chemically Reacting Systems

Simon L. Cotter,[1, a)] Konstantinos C. Zygalakis,[1] Ioannis G. Kevrekidis,[2] and Radek Erban[1, b)]
[1)] *Mathematical Institute, University of Oxford, 24-29 St. Giles', Oxford OX1 3LB United Kingdom*
[2)] *Department of Chemical Engineering and Program in Applied and Computational Mathematics, Princeton University, Princeton, NJ 08544 USA*

Abstract. Stochastic simulation of coupled chemical reactions is often computationally intensive, especially if a chemical system contains reactions occurring on different time scales. In this paper we introduce a multiscale methodology suitable to address this problem. It is based on the Conditional Stochastic Simulation Algorithm (CSSA) which samples from the conditional distribution of the suitably defined fast variables, given values for the slow variables. In the Constrained Multiscale Algorithm (CMA) a single realization of the CSSA is then used for each value of the slow variable to approximate the effective drift and diffusion terms, in a similar manner to the constrained mean-force computations in other applications such as molecular dynamics. We then show how using the ensuing Stochastic Differential Equation (SDE) approximation, we can in turn approximate average switching times in stochastic chemical systems.

I. INTRODUCTION

Stochastic simulation algorithms (SSAs) for systems of chemically reacting species have recently been used in modelling of a number of biological systems, including the phage λ lysis-lysogeny decision circuit[3], circadian rhythms[39] and cell cycle[26]. The most widely used SSA was developed by Gillespie[20] in 1970s. It simulates every single chemical reaction, and samples exactly from the solution to the corresponding chemical master equation. In order to characterise the behaviour of the system, one has to simulate a large number of reactions and simulations can become computationally intensive. For suitable classes of chemically reacting systems, one can sometimes use exact algorithms which are equivalent to the Gillespie SSA, and which are less computationally intensive. Examples include the Gibson-Bruck SSA[19] and the Optimized Direct Method[7]. Another approach, which does not introduce errors in simulations, is the use of parallel computing, for example on Graphics Processing Units[29,30].

The SSAs mentioned so far are equivalent to the Gillespie SSA, *i.e.*, they stochastically simulate the occurrence of every chemical reaction. This can be computationally very expensive, especially for systems involving abundant species. Also, often a modeller is not interested in every single reaction which takes place in the system. There can be multiple time scales, meaning that one has to simulate a large number of reactions while a modeller is interested in a more slowly evolving quantity[14].

One approach to accelerate the Gillespie SSA is the τ-leap method[22]. In this algorithm the number of reactions over a specified time increment τ is sampled from a Poisson distribution for each reaction channel. The problem with using Poisson random variables however, is that it is possible for a species' population to become negative. One possible way around this problem is the binomial τ-leap algorithm[36] which replaces the Poisson distribution with a binomial. Another way to tackle this problem is the so called R-leap method[4], which fixes the number of reaction firings, instead of the time interval τ, as in the τ-leap method.

Alternatively we can take a coarse grained approach where instead of treating the molecular populations as discrete random variables we can describe them in terms of their concentrations that change continuously. This can be achieved by the Chemical Langevin equation[21], which is a stochastic differential equation (SDE) acting like a bridge between the discrete stochastic simulation algorithm and the deterministic reaction rate equations. This approach, and its recent reformulations[32], may be less computationally intensive than the exact method. However, for some chemical systems, the CLE might lead to negative populations because its solutions have a small but positive probability of becoming negative[1,40].

The two different approaches described above do not take explicitly into account the separation of time scales. To explain the algorithms that do, we first formulate the multiscale problem in mathematical terms.

Let us consider a well-mixed thermally-equilibrated chemical system, comprising of N different chemical species $\{X_1, \dots, X_N\}$ interacting through M reaction channels $\{R_1, \dots, R_M\}$. The state of the system is given by $\mathbf{X}(t) = [X_1(t), \cdots, X_N(t)]$ where $X_i(t)$, $i = 1, \dots, N$, is the number of molecules of type X_i in the system at time t, where for simplicity we denote both the chemicals and the number of that chemical by the same symbol.

Following Cao et al[5], we assume that the reactions can be divided into two groups; *fast reactions* and *slow reactions*. Fast reactions are those which occur many times on a timescale for which it is very unlikely that the slow reactions will occur at all. An example of such a system with slow and fast reactions is given in Section II. Cao et al[5] define a *fast species* to be any species whose population gets changed by some fast reaction. Otherwise they are defined to be *slow species*. In this paper, we denote

[a)] Electronic mail: cotter@maths.ox.ac.uk
[b)] Electronic mail: erban@maths.ox.ac.uk

by $\mathbf{F}$ the fast species and by $\mathbf{S}$ the slow species. Together, the components of the fast and slow species form a basis for the state space of the system, *i.e.* the dimension of this basis is equal to the number of linearly independent species (taking into account possible conservation relations).

Algorithms for systems with fast and slow species have been developed in the literature by a number of authors. Haseltine and Rawlings[24] choose to simulate the fast reactions using Langevin dynamics while simulating the slow reactions using the Gillespie algorithm. This approach requires not only the time scale separation but also the volume of the system to be sufficiently large. However, this can be avoided by using the probability densities of the fast species conditioned on the slow species to estimate the effective propensity functions of the slow species[5,6,10]. Other types of solvers have also been put forward and analysed for differing timescales of interest[34,38].

The above approaches build algorithms which can simulate the evolution of slow variables while avoiding explicit simulation of all fast reactions. Another approach to multiscale computation is to estimate the probability distribution of slow variables[14]. The main idea is to use short bursts of appropriately initialised stochastic simulations to estimate coefficients of unavailable macroscopic coarse-grained equations[28]. This approach has been successfully applied to number of applications, including models in materials science,[23], cell motility[15] and social insect behaviour[41]. In the case of chemical systems, short bursts of stochastic simulations are used to estimate drift and diffusion coefficients for an approximating Fokker-Planck equation written in terms of the slow variables[14].

Constrained dynamics approaches have been used to approximate the drift term in other scenarios, particularly with regards to stochastic differential equations relating to molecular dynamics. "Umbrella" sampling[33,37] in molecular dynamics constrains the system by adding deep potential wells to the constrained variables so that the system never moves far away from its target value. "Blue moon" sampling[8] has also been developed to sample rare or long-time events. Naive constraining of dynamics in the MD scenario can lead to bias, and so different numerical schemes for the simulation of constrained SDEs have also been proposed[31].

In this paper, we introduce the Conditional Stochastic Simulation Algorithm (CSSA). This is an algorithm that allows us to sample from the distribution of the fast variables conditioned on the slow ones. The effective dynamics of the slow process can then be approximated using the conditional distribution either as a Markov jump process[5,6,10] or as a SDE[14]. Our approach is a coarse-grained one, *i.e.*, we choose to approach the effective dynamics as a SDE. We estimate the coefficients of the effective SDE on the fly during the sampling of the conditional distribution in the spirit of other multiscale approaches[38].

The paper is organised as follows. In Section II, we introduce an illustrative example which is used throughout the paper. The Conditional Stochastic Simulation Algorithm (CSSA) is presented in Section III. In Section IV, we introduce the Constrained Multiscale Algorithm (CMA), based on the CSSA, allowing us to constrain the Gillespie algorithm to one particular state of the slow variables. At the same time, the algorithm estimates the drift and diffusion terms of that slow variable, analogously to the constrained mean force calculations that some have used in molecular dynamics simulations[9]. We show that it sufficient to use a single realization of the CSSA for each value of the slow variable $\mathbf{S}$ for which we wish to estimate the drift and diffusion coefficients. Illustrative numerical results for the illustrative example from Section II are presented in Section V. In Section VI, we show how these methods can be used to explore how the dynamics of the system change as we vary the parameters, including probability density and average switching time in a bistable system. Finally, in Section VII, we briefly discuss issues regarding the length of our simulations with respect to the accuracy in our estimates of the effective dynamics of the slow variable.

II. AN ILLUSTRATIVE EXAMPLE

Our goal is to introduce a general multiscale algorithm for stochastic simulation of N chemical species X_i, $i = 1, 2, \dots, N$, which are subject to M reaction channels R_j, $j = 1, 2, \dots, M$. We will explain it using the following illustrative example

$$\emptyset \xrightarrow{k_1} X_1 \underset{k_3}{\overset{k_2}{\rightleftarrows}} X_2 \xrightarrow{k_4} X_3 \xrightarrow{k_5} \emptyset. \qquad (1)$$

In what follows, we label the reactions $R_1, \dots, R_5$ corresponding to the reaction rate subscripts. Each reaction R_j has associated with it a propensity function[20] $\alpha_j(t)$. It is defined so that the product $\alpha_j(t)\mathrm{d}t$ is the probability that the j-th reaction occurs in the infinitesimally small time interval $[t, t + \mathrm{d}t)$. In the case of model (1), the propensity functions are defined by

$$\alpha_1(t) = k_1 V, \quad \alpha_2(t) = k_2 X_1(t), \quad \alpha_3(t) = k_3 X_2(t),$$
$$\alpha_4(t) = k_4 X_2(t), \quad \alpha_5(t) = k_5 X_3(t),$$

where V is the volume of the reactor. Depending on the choice of parameters, this system can have several separated time scales. We consider this system with the following set of non-dimensionalised parameters:

$$k_1 V = 100, \quad k_2 = k_3 = 200, \quad k_4 = k_5 = 1. \qquad (2)$$

Note that the time variable is dimensionless. In Figure 1, we plot the time evolution of the different species of system (1), along with $(X_1 + X_2)/2$, starting from $X_1(0) = X_2(0) = X_3(0) = 100$. We see that X_1 and X_2 are changing quickly, while X_3 is changing slowly. However, note that if we consider the variable $X_1 + X_2$, this

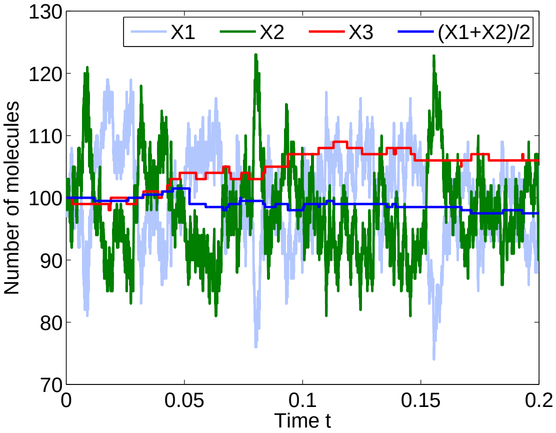

FIG. 1. *Sample trajectories of all three of the chemical species from the system* (1). *Parameters are given by* (2).

too is changing slowly. In this situation, reactions R_2 and R_3 are occurring on a faster time scale than the reactions R_1, R_4 and R_5, as can be seen in Table I, which shows the relative number of occurrences of each reaction. Therefore R_1, R_4 and R_5 are the slow reactions, and R_2 and R_3 are the fast reactions[5]. In Table II, we show changes of each of the listed variables during a standard Gillespie simulation of the system up to time $T = 10^4$.

On average, during this run, X_1 and X_2 changed over 200 times more often than either X_3 or $X_1 + X_2$. Given such a separation of time scales, slow variables are linear combinations of the chemical species, which are invariant with respect to the fast reactions. In this simple first-order chemical system, the two quantities X_3 and $X_1 + X_2$ are both invariant with respect to R_2 and R_3, and so are slow quantities. X_1 and X_2 by themselves are fast variables, as they change with every occurrence of reaction R_2 or R_3.

There are several different choices of bases of slow and fast variables for the system. In what follows we consider the following three examples:

$$S = [X_3], \qquad \mathbf{F} = [X_1, X_2]; \tag{3}$$
$$\mathbf{S} = [X_1 + X_2, X_3], \qquad F = [X_1]; \tag{4}$$
$$\mathbf{S} = [X_1 + X_2, X_3], \qquad F = [X_2]. \tag{5}$$

Notice that these systems have a different number of dimensions in the slow variable. In general, identifying which variables are slow and which are fast can be a challenging endeavour in itself, which is discussed in Section VIII.

A. Invariant measure of the illustrative example (1)

Since the chemical system (1) comprises only of monomolecular reactions, we can find explicitly the invariant measure of this system[25]. This result will be useful in Section V to demonstrate the accuracy of the CSSA and CMA. The invariant probability distribution on all of the variables is a multivariate Poisson distribution:

$$\begin{aligned}\mathbb{P}(X_1 = n_1, X_2 = n_2, X_3 = n_3)&\\ = \frac{\bar\lambda_1^{n_1}}{n_1!}\frac{\bar\lambda_2^{n_2}}{n_2!}\frac{\bar\lambda_3^{n_3}}{n_3!}\exp\left[-\left(\bar\lambda_1+\bar\lambda_2+\bar\lambda_3\right)\right],&\end{aligned} \tag{6}$$

where, using (2),

$$\bar\lambda_1 = \frac{k_1 V (k_3 + k_4)}{k_2 k_4} = 100.5,$$
$$\bar\lambda_2 = \frac{k_1 V}{k_4} = 100, \qquad \bar\lambda_3 = \frac{k_1 V}{k_5} = 100.$$

III. CONDITIONAL STOCHASTIC SIMULATION ALGORITHM (CSSA)

In this section, we introduce the Conditional Stochastic Simulation Algorithm (CSSA), an algorithm which samples from the conditional distribution $\mathbb{P}(\mathbf{F}|\mathbf{S} = \mathbf{s})$. This is the distribution of the fast variables $\mathbf{F}$ given a value $\mathbf{s}$ of the slow variables $\mathbf{S}$. This distribution is important for integrating correctly the effective dynamics of the slow variables, which is the ultimate aim of this paper. Given a state of the slow variables $\mathbf{S} = \mathbf{s}$, the CSSA is presented in Table III.

Initially we set the slow variable $\mathbf{S}$ to the value $\mathbf{s}$ for which we wish to sample $\mathbb{P}(\mathbf{F}|\mathbf{S} = \mathbf{s})$. We also initialise the fast variable $\mathbf{F} = \mathbf{f}$. Ideally we would take $\mathbf{f}$ to be a sample from $\mathbb{P}(\mathbf{F}|\mathbf{S} = \mathbf{s})$ itself. However, since this is the distribution we are actually interested in characterising, we can instead repeatedly run steps [**1**]-[**6**] until we are satisfied that the Markov chain has entered equilibrium and is drawing samples from $\mathbb{P}(\mathbf{F}|\mathbf{S} = \mathbf{s})$.

In step [**1**], as in the original Gillespie SSA, we calculate the values of the propensity functions. As the algorithm progresses, it is not necessary to recalculate all of these, as not all of them will change after each reaction has occurred. We can make use of the system's dependency graph[19] in order to ascertain which propensities are changed by each reaction. Steps [**2**] and [**3**] are the same as the original Gillespie algorithm. The waiting time to the next reaction is given by the following

Reaction	R_1	R_2	R_3	R_4	R_5
Frequency	0.25%	49.75%	49.50%	0.25%	0.25%

TABLE I. *List of percentages of occurrences of each reaction during a standard Gillespie simulation up to time* $T = 10^4$ *of the chemical system* (1).

Variable	X_1	X_2	X_3	$X_1 + X_2$
Changes	4.017×10^8	4.017×10^8	1.997×10^6	1.997×10^6

TABLE II. *List of numbers of changes of each of the listed variables during a standard Gillespie simulation up to time* $T = 10^4$ *of the chemical system* (1).

```
[1] Calculate propensity functions $\alpha_i(t)$, $i = 1, 2, \dots, M$.
[2] Next reaction time is given by (7).
[3] Choose one $j \in \{1, 2, \dots, M\}$, with probability $\alpha_j/\alpha_0$,
    and perform reaction $R_j$.
[4] If $\mathbf{S} \neq \mathbf{s}$ due to reaction $j$ occurring, then reset $\mathbf{S} = \mathbf{s}$
    while not changing the value of $\mathbf{F}$.
[5] If $X_i < 0$ for any $i$, then revert to the state of the
    system before the reaction $j$ occurred.
[6] Continue with step [1] with time $t = t + \tau$.
```

TABLE III. *Conditional Stochastic Simulation Algorithm (CSSA).*

formula:

$$\tau = -\frac{\log(u)}{\alpha_0}, \quad \text{where} \quad \alpha_0 = \sum_{i=1}^{M} \alpha_i(t), \tag{7}$$

and $u \sim U(0,1)$ is a sample from the uniform distribution on the unit interval $(0,1)$. The reaction R_j is chosen in step [**3**] using another $U(0,1)$ random variable.

Step [**4**] is where this algorithm diverges from the original Gillespie SSA. Here, if a reaction has occurred which changes the value of the slow variable $\mathbf{S}$, then we reset $\mathbf{S}$, thus constraining it to the desired value. Note that there is a unique way in how the reset is made, since the components of the variables $\mathbf{S}$ and $\mathbf{F}$ together form a basis of the state space, and we reset $\mathbf{S}$ in such a way that $\mathbf{F}$ is not altered.

Step [**5**] ensures that the states with negative copy numbers of one/more chemical species have zero probability mass. Without this step, the CSSA can run into difficulties if the reset of $\mathbf{S}$ causes one of the species to have a negative copy number. This step helps to inform our choice of fast variables, since some choices can naturally preserve this non-negativity condition without step [**5**], as we show in the next subsection.

A. Choice of fast and slow variables for the CSSA

In Section II we identified several candidates of fast and slow variables for the system (1). Let us consider the choice of variables (4). In this case, if reaction R_1 was to occur, the change in $\mathbf{S}$ is $[1,0]$. Since we must not alter the fast variable in step [**4**] of the CSSA, we must therefore reduce X_2 by one to reset $\mathbf{S}$. However, it is possible for reaction R_1 to occur when $X_2 = 0$, meaning that in this case X_2 could go below zero.

In many scenarios, however, these problems can be overcome by a better choice of fast variables, meaning that step [**5**] of the CSSA is no longer necessary. If we choose setup (5), then we are in just such a scenario, and this is certainly preferable. If, however, there was also a source reaction for X_2 (*i.e.* $\emptyset \longrightarrow X_2$), then no such choice of fast variables is possible, and step [**5**] is necessary.

IV. CONSTRAINED MULTISCALE ALGORITHM (CMA)

The CSSA described in Section III is able to sample from the probability distribution $\mathbb{P}(\mathbf{F}|\mathbf{S} = \mathbf{s})$. However, a modeller is often more interested in the effective dynamics of the slow variable $\mathbf{S}$. In this instance, we will attempt to approximate these dynamics by a continuous stochastic differential equation, which has corresponding Fokker-Planck equation given by

$$\frac{\partial p}{\partial t}(\mathbf{s}, t) = \nabla . \left(\nabla[D(\mathbf{s})\, p(\mathbf{s}, t)] - V(\mathbf{s})\, p(\mathbf{s}, t)\right). \tag{8}$$

The challenge now is to estimate the effective drift V and diffusion D of $\mathbf{S}$. In order to estimate this we can run the CSSA for a range of values of the slow variable. For each value $\mathbf{s}$ of $\mathbf{S}$, we use a CSSA simulation of length $T(\mathbf{s})$ to compute the following quantities:

$$V(\mathbf{s}) = \frac{1}{T(\mathbf{s})} \sum_{m=1}^{Q(T(\mathbf{s}))} \delta\mathbf{S}_m, \tag{9}$$

$$D(\mathbf{s}) = \frac{1}{2T(\mathbf{s})} \sum_{m=1}^{Q(T(\mathbf{s}))} \delta\mathbf{S}_m \delta\mathbf{S}_m^T, \tag{10}$$

where $d\mathbf{S}_m$ is the change in $\mathbf{S}$ due to the m-th iteration of the CSSA before the reset is made in step [**4**], $T(\mathbf{s})$ is the chosen length of CSSA simulation, and $Q(T(\mathbf{s}))$ is the number of iterations of the CSSA that are made up to time $T(\mathbf{s})$. These quantities are estimators for the effective drift V and diffusion D of the slow variable $\mathbf{S}$. The formulae (9) and (10) are derived in Appendix A.

Having estimated $V(\mathbf{s})$ and $D(\mathbf{s})$ by (9) and (10), we compute the approximation π_{CMA} of the invariant measure of the slow variables by solving

$$\nabla . \left(\nabla[D(\mathbf{s})\, \pi_{\text{CMA}}(\mathbf{s})] - V(\mathbf{s})\, \pi_{\text{CMA}}(\mathbf{s})\right) = 0, \tag{11}$$

which is the steady state equation corresponding to (8). Equation (11) is solved by standard numerical approaches[12]. It is worth noting that for a one dimensional slow variable, the solution to (11) can also be given analytically.

Since the calculation (9) and (10) of the drift and diffusion terms boil down to two different statistics on the size/number of resets of $\mathbf{S}$, the additional cost to estimate these quantities is negligible in comparison to the cost of running the CSSA. Furthermore, every single reaction that occurs which changes $\mathbf{S}$ (and therefore requires a reset of this variable in step [**4**] of the CSSA) has an effect on the final output, so that we are garnering all the possible information that is available from the stochastic simulation.

The length of simulation $T(\mathbf{s})$ is in general state dependent. We will discuss this state-dependence in Section VII. Another issue which will have an impact on the efficiency of the method is the choice of initialisation of the fast variable in step [**1**] of the CSSA. One might expect, for well behaved systems, the distributions $\mathbb{P}(\mathbf{F}|\mathbf{S} = \mathbf{s})$

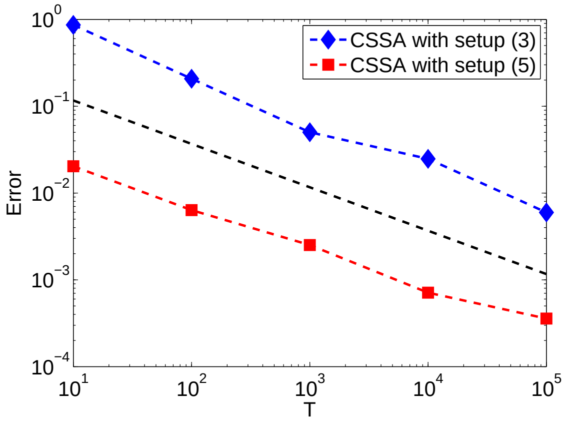

FIG. 2. *Error curve of the approximation of* $\mathbb{P}(\mathbf{F}|\mathbf{S}=\mathbf{s})$ *by the CSSA with both choices of variables* (3) *and* (5). *Black dashed line denotes decay proportional to* $1/\sqrt{T}$.

and $\mathbb{P}(\mathbf{F}|\mathbf{S}=\mathbf{s}+\boldsymbol{\varepsilon})$ would be very similar for small $\boldsymbol{\varepsilon}\in\mathbb{Z}^d$ where $d\in\mathbb{N}$ is the dimension of the slow variable $\mathbf{S}$. Therefore, once we have characterised the distribution $\mathbb{P}(\mathbf{F}|\mathbf{S}=\mathbf{s})$ (using, for example, an initial condition $\mathbf{F}$ from a "burn-in"), we could use a sample from this distribution to initialise the fast variable for the CSSA for the neighbouring values of $\mathbf{s}$. This is in some sense similar to the numerical continuation method, which is used to compute steady states in bifurcation analysis.

V. APPLICATION OF CMA TO MODEL (1)

In this section, we apply the CSSA and CMA to the illustrative example (1) presented in Section II. First we show that the CSSA is sampling correctly from the conditional distribution $\mathbb{P}(\mathbf{F}|\mathbf{S}=\mathbf{s})$. Since we have the explicit analytical steady state solution (6) of the chemical master equation, we can check the answers that we get using the CSSA (to ascertain the distribution of the fast variables). In particular, we have an explicit formula for $\mathbb{P}(\mathbf{F}|\mathbf{S}=\mathbf{s})$ which can be used in the error computation defined by:

$$\text{Error} \;=\; \sum_{\mathbf{f}\in\mathbb{N}^{3-d}} \left|\mathbb{P}(\mathbf{F}=\mathbf{f}|\mathbf{S}=\mathbf{s})-\nu_{\text{CSSA}}(\mathbf{f})\right|, \qquad (12)$$

where ν_{CSSA} is the conditional probability distribution estimated using the CSSA, and $d\in\mathbb{N}$ is the dimension of the slow variable vector $\mathbf{S}$.

In Figure 2, we present estimations of error of the CSSA algorithm for two choices of fast and slow variables given by (3) and (5). We run simulations for a fixed value of $T(\mathbf{s})$ and compute the error using (12). As can be seen in comparison with the black dashed line (which has decay proportional to $1/\sqrt{T}$), both of the choices of slow variable leave us with error which is approximately $\mathcal{O}(1/\sqrt{T})$. This is to be expected, since this is the rate of decay of the standard Monte Carlo error. However, the error incurred by using (3) is consistently larger than that for

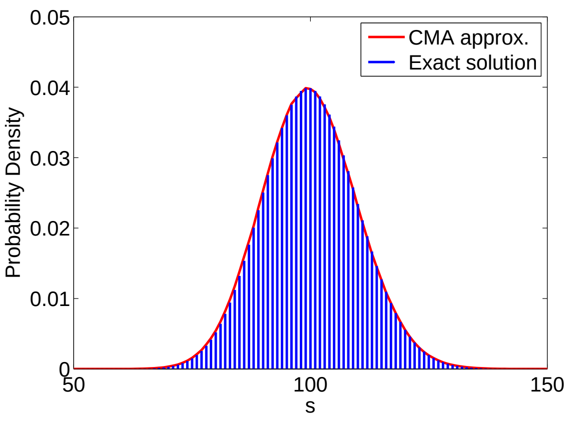

FIG. 3. *Approximation* π_{CMA} *of the probability density of* $S=X_3$ *by the CMA. Drift and diffusion functions are estimated from CMA runs with* 10^5 *changes in* $\mathbf{S}$ *for each* $\mathbf{s}$. *The blue histogram is an exact solution of the stationary chemical master equation, given by* (13).

(5), as there exists an alternative basis of $[X_1,X_2]$, given by $[X_1,X_1-X_2]$, which has a slow and a fast variable.

Next, we apply the CMA to the illustrative model (1), for both of the choices for the fast and slow variables, given by (3) and (5).

A. One-dimensional slow variable: Choice of variables (3)

The slower rate of convergence of the CSSA shown in Figure 2 due to the poor choice of slow/fast variables will naturally have a knock-on effect on the rate of convergence of the CMA approximation of the effective diffusion of the slow variable $S=X_3$. However, we can still accurately approximate the effective dynamics of the slow variable.

In Figure 3, we present π_{CMA} computed by (11). In this case, the time $T(s)$ is defined implicitly as the time it takes for L changes to occur in S for a given value of s, where we choose $L=10^5$. Since we have the expression (6) for the whole of the probability mass function, we can integrate out the fast variable $\mathbf{F}=[X_1,X_2]$, leaving us with the marginal probability distribution of the slow variable $S=X_3$, given by

$$\begin{aligned}\mathbb{P}(S=s) &= \sum_{f_1,f_2=0}^{\infty}\mathbb{P}(X_1=s,X_2=f_1,X_3=f_2)\\ &= \frac{100^s}{s!}\exp\left(-100\right). \qquad (13)\end{aligned}$$

This formula is plotted in Figure 3 as the blue histogram, and the comparison is excellent.

In Figure 3, we defined the time $T(s)$ implicitly by counting up to L changes in S. In Figure 4, we show how the results vary with L. We plot the error of this method as

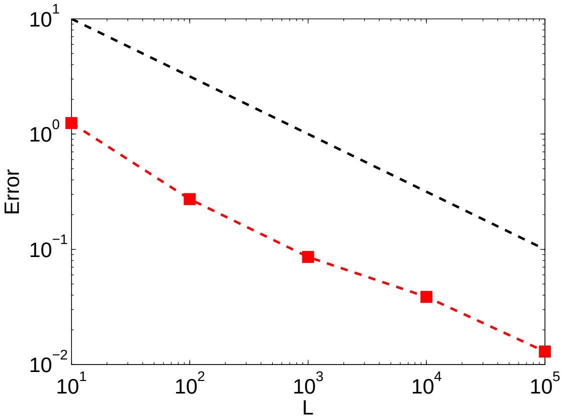

FIG. 4. *Error of the CMA, given by* (14)*, as a function of simulation length* L*. Black dashed line denotes decay proportional to* $1/\sqrt{T}$.

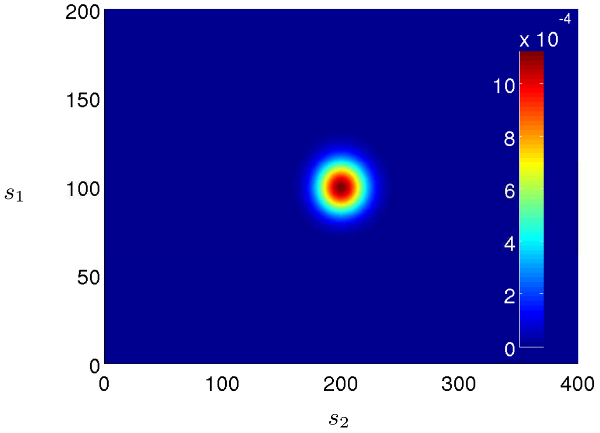

FIG. 5. *Approximation* π_{CMA} *of* $\mathbb{P}(\mathbf{S} = [s_1, s_2] = [X_1 + X_2, X_3])$ *by the CMA where we used CSSA simulations of length* $T(\mathbf{s}) \equiv 10^3$.

a function of L. We define the error to be:

$$\text{Error } = \sum_{s\in\mathbb{N}} \big|\mathbb{P}(S=s) - \pi_{\text{CMA}}(s)\big|. \tag{14}$$

Figure 4 shows that the CMA is approximately converging at a rate $\mathcal{O}(1/\sqrt{L})$, as in the CSSA examples earlier in this section. We will return to this issue in more detail in Section VII, where we look at how many changes in S are required for a given order of accuracy of CSSA as a function of the variable S.

B. Two-dimensional slow variable: Choice of variables (5)

In this case, the slow variable is two dimensional, which means that (11) must be approximated using numerical methods, such as finite elements[12]. Figure 5 shows π_{CMA} computed by solving (11) in domain $[0, 400]\times[0, 200]$ with no flux boundary conditions. Here we use a fixed length of simulation $T(\mathbf{s}) = 10^3$ for each value $\mathbf{s} = [s_1, s_2]$ of $\mathbf{S}$. The resulting two dimensional probability density function is indistinguishable from the steady state solution of the chemical master equation, given by:

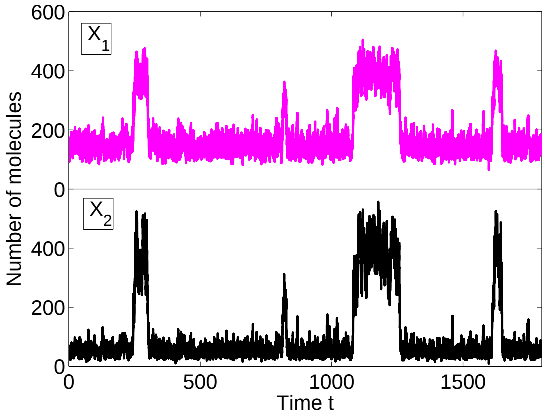

FIG. 6. *Trajectories of the chemical system* (16)*, showing switching between two favourable states, where the system parameters are given by* (17).

$$\mathbb{P}(X_1 + X_2 = s_1, X_3 = s_2) = \sum_{a=0}^{s_1} \frac{100^{s_1+s_2}201^a}{a!(s_1-a)!s_2!200^a} \exp\left(-300.5\right). \tag{15}$$

The disadvantage of this setup (5) in comparison with setup (3), is that if we wish to retain the same grid size for our approximation of the functions D and V as we did before, then the number of points for which we need to run the CSSA is squared. This is alleviated by the faster convergence for each value $\mathbf{s}$ of $\mathbf{S}$ as demonstrated in Figure 2.

VI. STOCHASTIC "BIFURCATION DIAGRAMS"

In this section, we show how the CMA can aid us in understanding how the behaviour of the slow variables change as we alter some system parameters. We will illustrate this using a simple bistable chemical system written in terms of two chemical species X_1 and X_2:

$$X_2 \underset{k_2}{\overset{k_1}{\rightleftarrows}} X_1 + X_2, \qquad \emptyset \underset{k_4}{\overset{k_3}{\rightleftarrows}} X_1, \qquad X_1 + X_1 \underset{k_6}{\overset{k_5}{\rightleftarrows}} X_2. \tag{16}$$

As before, we denote the reactions R_1, R_2, ..., R_6 according to the reaction rate subscripts. The propensity functions of reactions (16) are given by:

$$\begin{aligned} \alpha_1(t) &= k_1 X_2(t), & \alpha_2(t) &= k_2 X_1(t) X_2(t)/V, \\ \alpha_3(t) &= k_3 V, & \alpha_4(t) &= k_4 X_1(t), \\ \alpha_5(t) &= k_5 X_1(t)(X_1(t)-1)/V, & \alpha_6(t) &= k_6 X_2(t), \end{aligned}$$

where V is the system volume. In Figure 6, we show an illustrative trajectory of the system (16) computed with

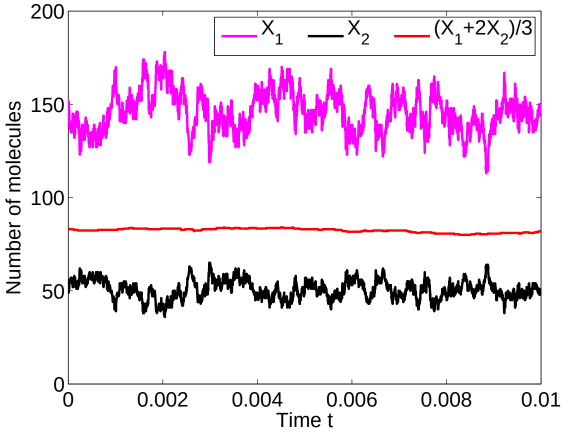

FIG. 7. *Trajectories of the chemical system* (16)*, showing slow behaviour of the variable* $S = X_1 + 2X_2$*, where the system parameters are given by* (17)*.*

the Gillespie algorithm, for dimensionless parameters:

$$\begin{array}{lll} k_1 = 32, & k_2/V = 0.04, & k_3 V = 1475, \\ k_4 = 19.75, & k_5/V = 10, & k_6 = 4000. \end{array} \qquad (17)$$

It is clear from these trajectories that the system has two favourable states around $[X_1, X_2] = [142, 50]$ and $[X_1, X_2] = [393, 386]$. In the parameter regime (17), reactions R_5 and R_6 are occurring on a much faster timescale than the other reactions. This leads us to the choice of slow variable $S = X_1 + 2X_2$ which is invariant with respect to these reactions, as confirmed in Figure 7. Choosing $F = X_2$ means that the CSSA naturally preserves the non-negativity of all of the chemical species of the system, *i.e.* step [**5**] of the CSSA in Table III is not needed.

A. CMA for the bistable example

In Figure 8, we plot π_{CMA} computed for the parameter values (17), with CSSA simulations of length $T(s) = 10^3$ for each value s of the slow variable $S = X_1 + 2X_2$. To demonstrate the accuracy of the CMA, we can compare π_{CMA} with the solution to the stationary chemical master equation. This can be approximated by making the assumption that given a big enough domain $\Omega = \{[X_1, X_2] \in [0, U_1] \times [0, U_2]\}$, the probability mass of the invariant distribution outside of Ω is negligible. Let us denote by $p_{n,m}(t)$ the probability that $[X_1, X_2] = [n, m]$ at time t. It evolves according to the chemical master equation for system (16), which is given by:

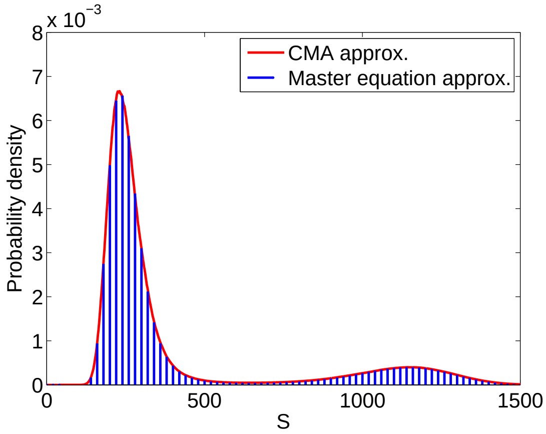

FIG. 8. *The estimated probability density of* $S = X_1 + 2X_2$*, both by the CMA (red solid line, using CSSA simulations of length* $T = 10^3$ *for each value* s *of* S*), and by approximation of the steady state of the chemical master equation* (18)*, on* $\Omega = [0, 700] \times [0, 1225]$ *(blue bars).*

$$\begin{aligned} \frac{\mathrm{d}p_{n,m}}{\mathrm{d}t} &= \Big[k_1 m + k_3 V\Big] p_{n-1,m} + k_6(m+1) p_{n-2,m+1} \\ &+ \Big[k_2(n+1)m + k_4(n+1)\Big] p_{n+1,m} \\ &+ k_5(n+2)(n+1) p_{n+2,m-1} - \Big[k_1 m \\ &+ k_2 nm + k_3 V + k_4 n + k_5 n(n-1) + k_6 m\Big] p_{n,m}. \end{aligned} \qquad (18)$$

Letting $p_{n,m} = 0$ for all $[n, m] \notin \Omega$, and setting the left hand side of (18) to zero, we are left with a linear system. Finding the eigenvector of the corresponding matrix, with eigenvalue zero, gives us an approximate solution of the stationary chemical master equation. Summing over the fast variable, we then obtain the probability distribution on the slow variable S, which is plotted in Figure 8 as the blue bars. The CMA approximation is visually indistinguishable from the approximation of the stationary CME.

B. Analysis of parameter dependence

One motivation for computing estimates of the effective dynamics of the slow variable, is to try to understand how the behaviour of the system is altered by a change in reaction rate parameters. In this section, we show that our SDE approximation of the effective dynamics of the slow variable can be used to compute a stochastic version of a bifurcation diagram for system (16). In the deterministic case (using mass action kinetics), the chemical system (16) is described by a system of two ordinary differential

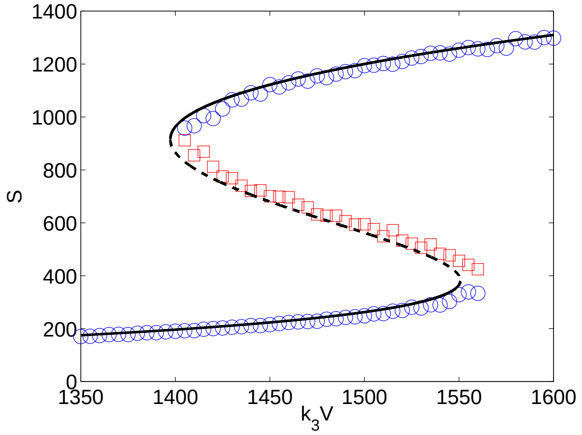

FIG. 9. *Stochastic bifurcation of the chemical system* (16) *with parameters given by* (17) *with* $k_3V \in [1350, 1600]$*, estimated using the CMA with length of simulation* $T = 10$*. Blue circles (red squares) denote the states which are estimated to be located at local maxima (minima) of probability density. The solid (dashed) lines denote stable (unstable) steady states of the ODE mean field approximation* (20)*.*

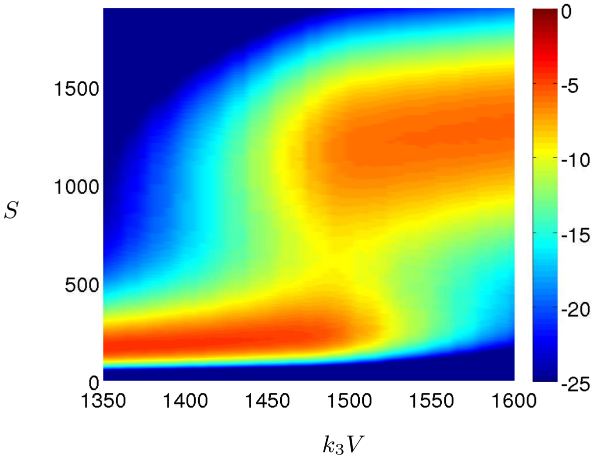

FIG. 10. *Log probability density of the steady state of the chemical system* (16) *with parameters given by* (17) *with* $k_3V \in [1350, 1600]$*, estimated using the CMA with length of simulation* $T = 10$*.*

equations (ODEs):

$$\frac{\mathrm{d}x_1}{\mathrm{d}t} = k_1x_2 - k_2x_1x_2 + k_3 - k_4x_1 - 2k_5x_1^2 + 2k_6x_2,$$
$$\frac{\mathrm{d}x_2}{\mathrm{d}t} = k_5x_1^2 - k_6x_2, \tag{19}$$

where x_1 and x_2 are concentrations of chemical species X_1 and X_2, respectively. To make a better comparison with the stochastic model, we introduce variables $\bar{X}_1 = x_1V$ and $\bar{X}_2 = x_2V$, which are approximations of the number of molecules of chemical species X_1 and X_2 as predicted by the ODE system (19). Multiplying (19) by V, we obtain the following system of ODEs for $\bar{X}_1$ and $\bar{X}_2$:

$$\frac{\mathrm{d}\bar{X}_1}{\mathrm{d}t} = k_1\bar{X}_2 - \frac{k_2}{V}\bar{X}_1\bar{X}_2 + k_3V - k_4\bar{X}_1 - 2\frac{k_5}{V}\bar{X}_1^2 + 2k_6\bar{X}_2, \tag{20}$$
$$\frac{\mathrm{d}\bar{X}_2}{\mathrm{d}t} = \frac{k_5}{V}\bar{X}_1^2 - k_6\bar{X}_2.$$

Let us consider the system (16) with parameters given by (17), but now with $k_3V \in [1350, 1600]$. In this interval, the mean field ODE approximation (20) of the dynamics of the system undergoes two fold bifurcations as it is shown in the corresponding bifurcation diagram plotted in Figure 9, denoted by the black line. Here a solid line indicates a stable steady state, and the dashed line indicates an unstable steady state. To make a direct comparison with the stochastic model we visualise the bifurcation diagram using the variable $S = \bar{X}_1 + 2\bar{X}_2$.

In the stochastic model, we can identify the favourable states by the local maxima of the probability density. To estimate this, we use the CMA with simulation length $T = 10$ for each parameter value k_3V, for each value of the slow variable $S = X_1 + 2X_2$. For each parameter value $k_3V \in [1350, 1600]$, we estimate π_{CMA}, as in Figure 8. The results for this are plotted in Figure 10, where the log density is plotted for each parameter value k_3V. We then estimate the positions of the local maxima in the density. The maxima are shown in Figure 9 as blue circles. For a one dimensional slow variable, as in this example, we can also identify local minima in a similar way, and these are plotted in Figure 9 as red squares. In this simple toy example the stochastic "bifurcation diagram" bears a strong resemblance to the bifurcation of the ODE approximation (20), however other systems exist in which they are vastly different[12].

Stochastic modelling also gives rise to additional phenomena which are not captured by mean field ODEs, such as the switching behaviour between the favourable states in this example. Therefore, in the stochastic case, we can also estimate the average length of time that it takes to go from one favourable state to the other, and then back to the first again. Having estimated drift and diffusion coefficients, one can obtain estimates for the average switching time, by solving a suitable elliptic partial differential equation[12], similar to (8). In the case of a one dimensional slow variable S, this equation can be solved explicitly[17]. Let us suppose that $s_1 < s_2$. To obtain the average time to reach state $S = s_2$, given that we started at state $S = s_1$:

$$\tau_{s_1 \to s_2} = 4\int_{s_1}^{s_2} \frac{1}{\log(\pi_{\text{CMA}}(y))}\int_0^y \frac{\log(\pi_{\text{CMA}}(z))}{D(z)}\mathrm{d}z\mathrm{d}y, \tag{21}$$

where π_{CMA} is the approximation of the density $\mathbb{P}(S = s)$. Similarly, the average time it takes for the process to reach state $S = s_1$, given that we started at state $S = s_2$,

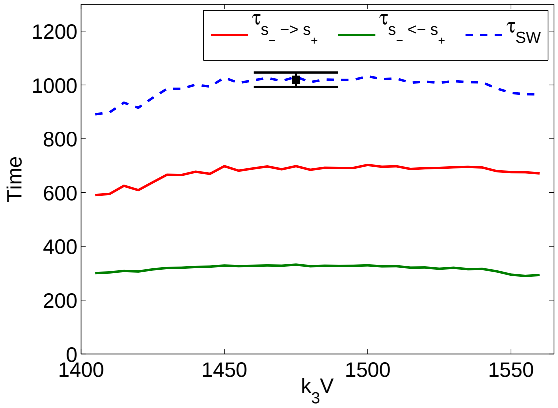

FIG. 11. *Estimated average switching time for the system* (16) *with parameters given by* (17) *with* $k_3V \in [1350, 1600]$*, using the CMA with length of simulation* $T(s) = 10$ *for each value* s *of* S*. Black plot denotes estimation using direct Gillespie estimation of the average switching time for* $k_3V = 1475$*. Error bar denotes one standard deviation (approximated by* $\sigma/\sqrt{10^3}$ *where* σ^2 *is the sample variance and the average was taken over* 10^3 *switches).*

is given by:

$$\tau_{s_1 \leftarrow s_2} = 4 \int_{s_1}^{s_2} \frac{1}{\log(\pi_{\text{CMA}}(y))} \int_{y}^{\infty} \frac{\log(\pi_{\text{CMA}}(z))}{D(z)} \mathrm{d}z\mathrm{d}y. \tag{22}$$

Let us denote the local minima of π_{CMA} by s_u (red squares in Figure 9), and local maxima of π_{CMA} by $s_- < s_+$ (blue circles in Figure 9). Figure 11 shows the plots for $\tau_{s_- \to s_+}$ and $\tau_{s_- \leftarrow s_+}$, computed by (21) and (22) respectively. Naturally we truncate the second integral in (22) to compute it numerically. A modeller is sometimes interested in an average "oscillation time", which can be computed as $\tau_{\text{SW}} = \tau_{s_- \to s_+} + \tau_{s_- \leftarrow s_+}$, *i.e.* the average time it takes to go from one state of locally maximal probability density to the other, and back again. This is plotted as the dashed line in Figure 11. Additionally, we plot on this figure the results of a direct computation of average switching time, with $k_3V = 1475$. This value was averaged over 1000 switches from one well to the other, using the Gillespie algorithm. For this parameter set, the agreement between the CMA approximation of the switching time and the direct stochastic computations is remarkable.

The formulae (21) and (22) can be used for general estimation of average travel times between two points for one dimensional slow variables. For example in Figure 12 we show the average exit times from each well which are defined by $\tau_{s_- \to s_u}$ and $\tau_{s_u \leftarrow s_+}$ respectively. If the slow variable is of more than one dimension, then no exact formula like (21) exists for the transition times, and the corresponding elliptic PDE needs to be solved numerically[12].

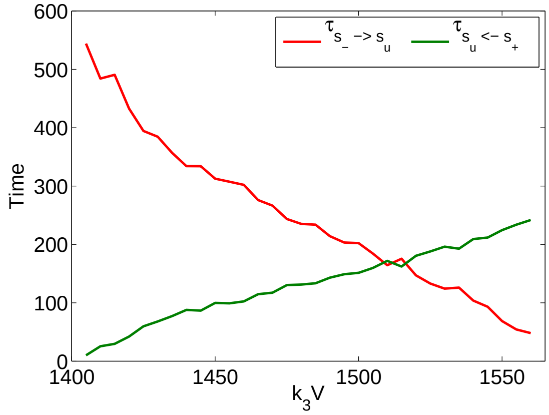

FIG. 12. *Estimated average escape time for the two potential wells in the system* (16) *with parameters given by* (17) *with* $k_3V \in [1350, 1600]$*, using the CMA with length of simulation* $T(s) = 10$ *for each value* s *of* S*.*

VII. CHOOSING A SUITABLE LENGTH OF SIMULATION

When implementing the CSSA or CMA, one has to choose the state-dependent length of simulation $T(\mathbf{s})$. There are three ways to do this. The simplest approach is to select a fixed $T(\mathbf{s}) \equiv \hat{T} \in \mathbb{R}^+$. Alternatively, one might also want to define $T(\mathbf{s})$ indirectly by postulating that we allow $L \in \mathbb{N}$ changes of the slow variable $\mathbf{S}$ before stopping the simulation for each value $\mathbf{s}$. The third possibility is to design a stopping condition within the algorithms, to determine when the estimation of the observable we are interested in has converged, up to defined tolerances. This may vary depending on what the aim of the simulation is, whether it be approximation of switching times, or of the probability density, or some other quantity. However a sensible stopping condition with the desired properties may be hard to identify. In this section we instead look at how choosing $T(\mathbf{s})$ in one of the first two ways, either setting it to be a constant, or through allowing only L changes of the slow variable, over all values $\mathbf{s}$ of $\mathbf{S}$, can affect the accuracy of the approximation of a given quantity.

Naturally, there is a linear relationship between the total cost of running the CMA, with a given fixed simulation length $T(\mathbf{s}) = \hat{T} \in \mathbb{R}^+$ (or fixed number of changes $L \in \mathbb{N}$ of the slow variable) for each value $\mathbf{s}$ of $\mathbf{S}$. This has been proven to be the case in our numerical simulations. In the following we measure computational cost $\mathbb{Q}$ in terms of how many individual reactions have been simulated in the algorithm. In the CMA, $Q(T(\mathbf{s}))$ denotes the total number of reactions simulated in the algorithm up to time $T(\mathbf{s})$. This implies that

$$\mathbb{Q} \equiv \mathbb{Q}(T(\cdot)) = \sum_{s} Q(T(\mathbf{s})),$$

where the sum is over the chosen region of estimation

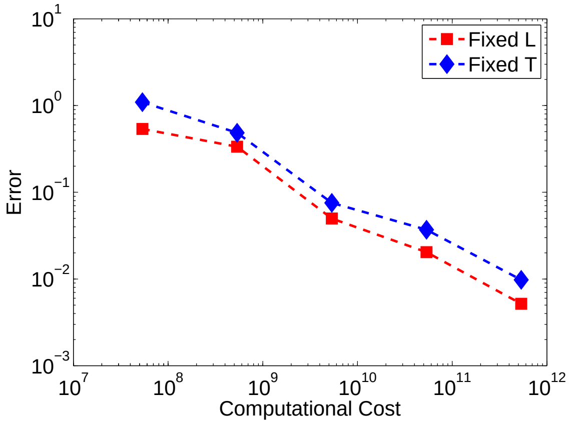

FIG. 13. *Error curves for fixed simulation length $\hat{T}$ and for fixed number of changes L of S for each value s, for the system* (16)*, with parameters given by* (17)*, as a function of total computational cost.*

of the slow variable in the algorithm. By running the algorithm for different values of $\hat{T}$ (or L in the case of limiting changes of the slow variable) and counting the total number of reaction evaluations, the linear relationship between simulation length parameter $\hat{T}$ (resp. L) and the total computational cost $\mathbb{Q}$ can then be found. We can then use these approximations to compare the calculated π_{CMA} for the system using a fixed $\hat{T}$ and L respectively, with roughly the same total computational cost, and then compare these approximations with the approximation to the solution of the stationary chemical master equation (18).

Figure 13 shows the errors (as defined by (14)) in the approximation π_{CMA} as a function of total computational cost, both for fixed simulation length $\hat{T}$, and for fixed number of changes L of S, for the bistable chemical system (16) with reaction parameters given by (17). In this example, the simulations with fixed number L of changes of the slow variable, instead of fixed $T(s) = \hat{T}$, led to approximately half the error in the resulting estimation π_{CMA} of the invariant distribution, when given the same amount of computational resource. This result is system-dependent however, since the same error curves computed for the illustrative system (1) gave results which were barely distinguishable for the two simplest methods of choosing the simulation length.

In theory, we could also estimate a suitable length of simulation by calculating the smallest non-zero eigenvalue of the evolution equation describing the system dynamics. This is equivalent to solving the chemical master equation, and thus is computationally expensive or in some cases unfeasible. For simple systems, such as (1), such a computation is possible. In particular, we find that the smallest negative eigenvalue for the choice of fast and slow variables (3) is smaller in magnitude than the one corresponding to the choice (5), which is consistent with error plots seen in Figure 2.

VIII. DISCUSSION

In this paper, we developed a multiscale methodology for analysis of systems with multiple timescales. We demonstrated that this approach can be used to approximate important characteristics of systems, such as invariant probability density and switching times. One of the limitations of the method is the dimension of the slow variable vector (*i.e.* number of slow variables). In this paper, we presented examples with one or two slow variables. In biological applications, biochemical systems can be written in terms of tens of variables[3,26,39], often reacting on different timescales. In order to apply this approach to such a system, we must first identify the slowly changing quantities. The direct exploration of a system, similar to that presented in Section II, might provide a guidance for the selection of the slow variable(s). However, in some systems, different variables are slow (in comparison with the other variables) in different parts of the state space. In this case, an automated analysis might be needed to identify the slow manifold[13,35]. If the dimension of the slow variables is only one or two, then the CMA can be applied exactly as we have here, no matter what the overall dimension of the system.

The situation becomes more challenging when the system has three or more slow variables, because of the high dimensional PDEs (8) which are computationally intensive to solve. To tackle this problem, we are currently investigating the most efficient approach for solution of these PDEs. This problem differs from typical problems in numerical analysis, in that we have to optimise the choice of domain in which the drift and diffusion coefficients are estimated. Further to this, since the estimations are made using stochastic simulations, they are noisy, and may need to be smoothed in order to obtain good results. We will report on our findings in the future.

In some biological examples, there are variables which can get down to single digit values (or even zero). For example, the relative abundance of some mRNA molecules in cells can be less than one, on average[2,26]. Gene states are sometimes represented as Boolean variables (a variable which is zero for a specified gene being "off", and equal to one when the gene is transcribed)[16]. In some cases, these variables are fast and do not cause any problems[27]. However, if a slow variable can only have a few discrete values, then the SDE-based approximations can lead to errors, and slow algorithms written in terms of Markov chains are more applicable[6,11]. To improve efficiency of this approach, one could also couple this with the effective diffusivity approach for areas with larger copy numbers, in a hybrid model. The CMA can also be modified by substituting the Gillespie algorithm by a statistically equivalent version, such as the Gibson-Bruck algorithm, using a suitable dependency graph[19].

Appendix A: Derivation of drift and diffusion formulae

Considering the Fokker-Planck equation (8), the drift V and diffusion D are given by the following expressions:

$$V(\mathbf{s}) = \lim_{\Delta t \to 0} \frac{\langle \mathbf{S}(\Delta t) - \mathbf{s} \,|\, \mathbf{S}(0) = \mathbf{s} \rangle}{\Delta t}, \qquad \text{(A1)}$$
$$D(\mathbf{s}) = \lim_{\Delta t \to 0} \frac{\big\langle [\mathbf{S}(\Delta t) - \mathbf{s}][\mathbf{S}(\Delta t) - \mathbf{s}]^T \,|\, \mathbf{S}(0) = \mathbf{s} \big\rangle}{2\Delta t},$$

where $\langle \cdot \rangle$ denotes the average with respect to the probability distribution $\mathbb{P}(\mathbf{F}|\mathbf{S} = \mathbf{s})$, which we can sample from using the CSSA in Section III. The CSSA which we introduced in the previous section is just one possible way to determine the conditional distribution on the fast variable given the slow variable. The advantage of this method over others[18], is that we are able to estimate the drift and diffusion terms of the slow variables on the fly, while sampling the conditional distribution of the fast variables.

Let us define $\delta\mathbf{S}_j$ to be the change in $\mathbf{S}$ due to the j-th iteration of the CSSA. For example, in the case of the illustrative example system (1) for the choice of slow variable $S = X_1 + X_2$, where the jth reaction to occur was R_i, we have:

$$\delta S_j = \begin{cases} 0 & i = 2, 3, 5, \\ 1 & i = 1, \\ -1 & i = 4. \end{cases}$$

We define the variable $\widetilde{\mathbf{S}}$, to be the sum of these elementary changes $\delta\mathbf{S}_j$, *i.e.*

$$\widetilde{\mathbf{S}}(t) = \sum_{j=1}^{Q(t)} \delta\mathbf{S}_j. \qquad \text{(A2)}$$

where $Q(t)$ iterations of the CSSA have been simulated up to some time $t > 0$. Since $\mathbf{S}(0) = \mathbf{s}$, formula (A1) implies that

$$V(\mathbf{s}) = \lim_{\Delta t \to 0} \frac{\langle \mathbf{S}(\Delta t) - \mathbf{s} | \mathbf{S}(0) = \mathbf{s} \rangle}{\Delta t} = \lim_{\Delta t \to 0} \frac{\big\langle \widetilde{\mathbf{S}}(t + \Delta t) - \widetilde{\mathbf{S}}(t) \big\rangle}{\Delta t},$$

for all $t > 0$, due to the resetting in step [**4**] of the CSSA, in Table III. Therefore, if $T = J\Delta t$ is the length of the CSSA simulation, we get

$$V(\mathbf{s}) = \lim_{\Delta t \to 0} \lim_{J \to \infty} \frac{1}{J} \sum_{j=1}^{J} \frac{\widetilde{\mathbf{S}}(j\Delta t) - \widetilde{\mathbf{S}}((j-1)\Delta t)}{\Delta t} = \lim_{\Delta t \to 0} \lim_{J \to \infty} \frac{1}{J\Delta t} (\widetilde{\mathbf{S}}(J\Delta t) - \widetilde{\mathbf{S}}(0)).$$

Using $\widetilde{\mathbf{S}}(0) = 0$, $T = J\Delta t$ and (A2), we have that

$$V(\mathbf{s}) = \lim_{T \to \infty} \frac{1}{T} \widetilde{\mathbf{S}}(T) = \lim_{T \to \infty} \frac{1}{T} \sum_{m=1}^{Q(T(\mathbf{s}))} \delta\mathbf{S}_m,$$

which is the formula given by (9). In a similar way, we can derive (10), namely

$$\begin{aligned} D(\mathbf{s}) &= \frac{1}{2} \lim_{\Delta t \to 0} \frac{\big\langle [\mathbf{S}(\Delta t) - \mathbf{s}][\mathbf{S}(\Delta t) - \mathbf{s}]^T | \mathbf{S}(0) = \mathbf{s} \big\rangle}{\Delta t} \\ &= \frac{1}{2} \lim_{J \to \infty} \lim_{\Delta t \to 0} \frac{1}{J\Delta t} \sum_{j=1}^{J} [\widetilde{\mathbf{S}}(j\Delta t) - \widetilde{\mathbf{S}}((j-1)\Delta t)] \\ &\qquad [\widetilde{\mathbf{S}}(j\Delta t) - \widetilde{\mathbf{S}}((j-1)\Delta t)]^T \\ &= \lim_{T \to \infty} \frac{1}{2T} \sum_{m=1}^{Q(T(\mathbf{s}))} \delta\mathbf{S}_m \delta\mathbf{S}_m^T. \end{aligned}$$

[1] D. Adalsteinsson, D. McMillen, and T.C. Elston. Biochemical network stochastic simulator (bionets): seoftware for stochastic modeling of biochemical networks. *BMC Bioinformatics*, 5(24), 2004.

[2] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular Biology of the Cell.* Garland Science, New York, 2008.

[3] A. Arkin, J. Ross, and H. McAdams. Stochastic kinetic analysis of developmental pathway bifurcation in phage l-infected Escherichia coli cells. *Genetics*, 149:1633–1648, 1998.

[4] A. Auger, P. Chatelain, and P. Koumoutsakos. R-leaping: Accelerating the stochastic simulation algorithm by reaction leaps. *Journal of Chemical Physics*, 125(8):084103, 2006.

[5] Y. Cao, D. Gillespie, and L. Petzold. Multiscale stochastic simulation algorithm with stochastic partial equilibrium assumption for chemically reacting systems. *Journal of Computational Physics*, 206:395–411, 2005.

[6] Y. Cao, D. Gillespie, and L. Petzold. The slow-scale stochastic simulation algorithm. *Journal of Chemical Physics*, 122(1):14116, 2005.

[7] Y. Cao, H. Li, and L. Petzold. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *Journal of Chemical Physics*, 121(9):4059–4067, 2004.

[8] G. Ciccotti, R. Kapral, and E. Vanden-Eijnden. Blue moon sampling, vectorial reaction coordinates, and unbiased constrained dynamics. *ChemPhysChem*, (6):1809–1814, 2005. DOI: 10.1002/cphc.200400669.

[9] G. Ciccotti, T. Lelievre, and E. Vanden-Eijnden. Projection of diffusions on submanifolds: application to mean force computation. *Comm. Pure Appl. Math.*, 61(3):371–408, 2008.

[10] W. E, D. Liu, and E. Vanden-Eijnden. Nested stochastic simulation algorithm for chemical kinetic systems with disparate rates. *Journal of Chemical Physics*, 123:194107, 2005.

[11] W. E, D. Liu, and E. Vanden-Eijnden. Nested stochastic simulation algorithms for chemical kinetic systems with multiple time scales. *J. Comput. Phys.*, 221(1):158–180, 2007.

[12] R. Erban, S.J. Chapman, I.G. Kevrekidis, and T. Vejchodský. Analysis of a stochastic chemical system close to a SNIPER bifurcation of its mean-field model. *SIAM J. Appl. Math.*, 70(3):984–1016, 2009.

[13] R. Erban, T. Frewen, X. Wang, T. Elston, R. Coifman, B. Nadler, and I. Kevrekidis. Variable-free exploration of stochastic models: a gene regulatory network example. *Journal of Chemical Physics*, 126:155103, 2007.

[14] R. Erban, I. Kevrekidis, D. Adalsteinsson, and T. Elston. Gene regulatory networks: A coarse-grained, equation-free approach to multiscale computation. *Journal of Chemical Physics*, 124:084106, 2006.

[15] R. Erban, I. Kevrekidis, and H. Othmer. An equation-free computational approach for extracting population-level behavior from individual-based models of biological dispersal. *Physica D*, 215(1):1–24, 2006.

[16] R. Erban, I.G. Kevrekidis, D. Adalsteinsson, and T.C. Elston. Gene regulatory networks: A coarse-grained, equation-free approach to multiscale computation. *Journal of Chemical Physics*, 124(8), 2006.

[17] C. W. Gardiner. *Handbook of stochastic methods*, volume 13 of *Springer Series in Synergetics*. Springer-Verlag, Berlin, second edition, 1985. For physics, chemistry and the natural sciences.

[18] A.E. Gelfand and A.F.M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):pp. 398–409, 1990.

[19] M. Gibson and J. Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *Journal of Physical Chemistry A*, 104:1876–1889, 2000.

[20] D. Gillespie. Exact stochastic simulation of coupled chemical reactions. *Journal of Physical Chemistry*, 81(25):2340–2361, 1977.

[21] D. Gillespie. The chemical Langevin equation. *Journal of Chemical Physics*, 113(1):297–306, 2000.

[22] D. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *Journal of Chemical Physics*, 115(4):1716–1733, 2001.

[23] M. Haataja, D. Srolovitz, and I. Kevrekidis. Apparent hysteresis in a driven system with self-organized drag. *Physical Review Letters*, 92(16):160603, 2004.

[24] E.L. Haseltine and J.B. Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *Journal of Chemical Physics*, 117(15):6959–6969, 2002.

[25] T. Jahnke and W. Huisinga. Solving the chemical master equation for monomolecular reaction systems analytically. *J. Math. Biol.*, 54(1):1–26, 2007.

[26] S. Kar, W. Baumann, M. Paul, and J. Tyson. Exploring the roles of noise in the eukaryotic cell cycle. *Proceedings of the National Academy of Sciences USA*, 16:6471–6476, 2009.

[27] T.B. Kepler and T.C. Elston. Stochasticity in transcriptional regulation: Origins, consequences, and mathematical representations. *Biophysical Journal*, 91:3116–3136, December 2001.

[28] I. Kevrekidis, C. Gear, J. Hyman, P. Kevrekidis, O. Runborg, and K. Theodoropoulos. Equation-free, coarse-grained multiscale computation: enabling microscopic simulators to perform system-level analysis. *Communications in Mathematical Sciences*, 1(4):715–762, 2003.

[29] G. Klingbeil, R. Erban, M. Giles, and P. Maini. Fat vs. thin threading approach on GPUs: application to stochastic simulation of chemical reactions. To appear in *IEEE Transactions on Parallel and Distributed Systems*, 2010.

[30] G. Klingbeil, R. Erban, M. Giles, and P. Maini. STOCHSIMGPU: Parallel stochastic simulation for the Systems Biology Toolbox 2 for MATLAB. To appear in *Bioinformatics*, 2010.

[31] T. Lelièvre, C. Le Bris, and E. Vanden-Eijnden. Analyse de certains schémas de discrétisation pour des équations différentielles stochastiques contraintes. *C. R. Math. Acad. Sci. Paris*, 346(7-8):471–476, 2008.

[32] B. Mélykúti, K. Burrage, and K.C. Zygalakis. Fast stochastic simulation of biochemical reaction systems by alternative formulations of the chemical langevin equation. *Journal of Chemical Physics*, 132(16):164109, 2010.

[33] B. Roux. The calculation of the potential of mean force using computer simulations. *Computer Physics Communications*, 91(1-3):275 – 282, 1995.

[34] H. Salis and Y. Kaznessis. Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions. *Journal of Chemical Physics*, 122(5):054103, 2005.

[35] A. Singer, R. Erban, I. Kevrekidis, and R. Coifman. Detecting the slow manifold by anisotropic diffusion maps. *Proceedings of the National Academy of Sciences USA*, 106(38):16090–16095, 2009.

[36] T. Tian and K. Burrage. Binomial leap methods for simulating stochastic chemical kinetics. *Journal of Chemical Physics*, 121(21):10356–10364, 2004.

[37] G.M. Torrie and J.P. Valleau. Nonphysical sampling distributions in monte carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, 23:187–199, 1977.

[38] E. Vanden-Eijnden. Numerical techniques for multi-scale dynamical systems with stochastic effects. *Commun. Math. Sci.*, 1(2):385–391, 2003.

[39] J. Villar, H. Kueh, N. Barkai, and S. Leibler. Mechanisms of noise-resistance in genetic oscillators. *Proceedings of the National Academy of Sciences USA*, 99(9):5988–5992, 2002.

[40] J. Wilkie and Y.M. Wong. Positivity preserving chemical langevin equations. *Chemical Physics*, 353:132–138, 2008.

[41] C. Yates, R. Erban, C. Escudero, I. Couzin, J. Buhl, I. Kevrekidis, P. Maini, and D. Sumpter. Inherent noise can facilitate coherence in collective swarm motion. *Proceedings of the National Academy of Sciences USA*, 106(14):5464–5469, 2009.

Acknowledgements The research leading to these results has received funding from the European Research Council under the *European Community's* Seventh Framework Programme (*FP7/2007-2013*)/ ERC *grant agreement* No. 239870. This publication was based on work supported in part by Award No KUK-C1-013-04, made by King Abdullah University of Science and Technology (KAUST). The work of I. G. K was partially supported by the US Department of Energy. Thanks also to Tomáš Vejchodský for useful conversations regarding the assembly and solution within the finite element method.