Stochastic chemical kinetics and the total quasi-steady-state assumption: Application to the stochastic simulation algorithm and chemical master equation

Shev MacNamara,[1,2,a)] Alberto M. Bersani,[3,b)] Kevin Burrage,[2,4,5,c)] and Roger B. Sidje[6,d)]
[1]*Department of Mathematics and The Australian Centre in Bioinformatics, The University of Queensland, Brisbane 4072, Australia*
[2]*The Institute for Molecular Biosciences, The University of Queensland, Brisbane 4072, Australia*
[3]*Department of Mathematical Methods and Models, "La Sapienza" University, Rome, Italy*
[4]*The Australian Centre in Bioinformatics, The University of Queensland, Brisbane 4072, Australia*
[5]*Oxford Computing Laboratory and Oxford Centre for Integrative Systems Biology, The University of Oxford, Oxford OX1 3QD, United Kingdom*
[6]*Department of Mathematics, The University of Alabama, P. O. Box 870350, Tuscaloosa, Alabama 35487-0350, USA*

(Received 10 May 2007; accepted 24 July 2008; published online 3 September 2008)

Recently the application of the quasi-steady-state approximation (QSSA) to the stochastic simulation algorithm (SSA) was suggested for the purpose of speeding up stochastic simulations of chemical systems that involve both relatively fast and slow chemical reactions [Rao and Arkin, J. Chem. Phys. **118**, 4999 (2003)] and further work has led to the nested and slow-scale SSA. Improved numerical efficiency is obtained by respecting the vastly different time scales characterizing the system and then by advancing only the slow reactions exactly, based on a suitable approximation to the fast reactions. We considerably extend these works by applying the QSSA to numerical methods for the direct solution of the chemical master equation (CME) and, in particular, to the finite state projection algorithm [Munsky and Khammash, J. Chem. Phys. **124**, 044104 (2006)], in conjunction with Krylov methods. In addition, we point out some important connections to the literature on the (deterministic) total QSSA (tQSSA) and place the stochastic analogue of the QSSA within the more general framework of aggregation of Markov processes. We demonstrate the new methods on four examples: Michaelis–Menten enzyme kinetics, double phosphorylation, the Goldbeter–Koshland switch, and the mitogen activated protein kinase cascade. Overall, we report dramatic improvements by applying the tQSSA to the CME solver. © *2008 American Institute of Physics*. [DOI: 10.1063/1.2971036]

I. INTRODUCTION

Chemical kinetics are often modeled by ordinary differential equations (ODEs) but under some circumstances—for example, when some species are present in small numbers[1,2]—a discrete and stochastic framework is more appropriate.[3] Such a framework is provided by the chemical master equation (CME),[3,4] which has been successfully used in systems biology to model gene regulatory networks as a collection of biochemical reactions. Intrinsic noise is known to be especially important in biological systems where small numbers of key regulatory molecules are often involved.[1,2] Models of the bacteriophage λ life cycle have been a flagship for the success of this approach.[5]

A very popular method for studying and simulating intrinsic noise is the stochastic simulation algorithm (SSA).[4,6] However, the SSA can become too slow in the presence of large molecular populations and/or large rate constants, thus motivating the (Poisson) τ-leap approximation,[7] accelerated leap methods,[8–10] and more generally, multiscale methods for simulating biochemical kinetics.[11,12] In the presence of both fast and slow reactions, the quasi-steady-state approximation (QSSA) has been one such multiscale method that has recently received much attention for the purpose of speeding up simulations of chemical reactions.[13–20] Here, we investigate its application to the direct solution of the CME, which describes the evolution of the probability mass function associated with the SSA. Significantly, we are able to adapt a CME solver, based on Krylov methods,[21–23] by incorporating a type of QSSA and thus take advantage of the multiscale nature of the systems being studied.

This paper is organized as follows. First, we discuss the mathematical framework of the CME and then give an analysis of how the QSSA is applied, distinguishing among different forms of the QSSA. Results of testing these new methods are reported and the strengths and limitations of the work are discussed.

A. Background to models of biochemical kinetics

The framework of the CME (Refs. 3 and 4) is now described. A biochemical system consists of N different kinds

[a)] Electronic mail: s.macnamara@imb.uq.edu.au.
[b)] Electronic mail: bersani@dmmm.uniroma1.it.
[c)] Electronic mail: kevin.burrage@comlab.ox.ac.uk and k.burrage@imb.uq.edu.au.
[d)] Electronic mail: roger.b.sidje@ua.edu.

of chemical species $\{S_1,\ldots,S_N\}$, interacting via M chemical reactions $\{R_1,\ldots,R_M\}$. It is assumed that the mixture has constant volume, is homogeneous, and that it is at thermal equilibrium. The system is modeled as a temporally homogeneous, continuous-time, discrete-state, Markov process. While macromolecular crowding effects leading to anomalous diffusion can be significant when describing processes on the membrane of a cell or within a cell,[24,25] this framework has proved to be successful in a number of biological settings.[5] The state of the system, $\mathbf{x}\equiv(x_1,\ldots,x_N)$, is a vector of non-negative integers where x_i is the number of molecules of species S_i. Transitions between states occur when a reaction occurs. Associated with each reaction R_j is a *stoichiometric* vector $\boldsymbol{\nu}_j$, of the same dimension as the state vector, that defines the way the state changes when a reaction occurs; if the system is in state $\mathbf{x}$ and reaction j occurs, then the system transitions to state $\mathbf{x}+\boldsymbol{\nu}_j$. Associated with each state is a set of M *propensities*, $\alpha_1(\mathbf{x}),\ldots,\alpha_M(\mathbf{x})$ that determine the relative chance of each reaction occurring. The propensities are defined by the requirement that, given $\mathbf{x}(t)=\mathbf{x}$, $\alpha_j(\mathbf{x})dt$ is the probability of reaction j occurring in the next infinitesimal time interval $[t,t+dt)$. They involve a "specific probability rate constant" c, which is measured in terms of probability per unit time and depends on microphysical properties of the molecules, temperature, and volume.[4] For numerical testing we may assume that the appropriate scalings have been taken care of and report the values of c and t that were used.

B. The SSA and leap methods

The SSA[4,6] simulates chemical systems one reaction at a time. At each step, it samples the waiting time until the next reaction occurs from an exponential distribution, and samples from a uniform distribution to determine the reaction number, based on the relative sizes of the propensity functions. However, as noted, it can become too slow in situations where some fast reactions are associated with very large propensity functions. The (Poisson) τ-leap approximation[7] speeds up the simulation by leaping forward through a much larger interval in time, with the number of times a reaction fires being drawn from the Poisson distribution. Following this idea, the midpoint τ-leap method,[7] implicit τ-leap method,[26] Poisson–Runge-Kutta method,[11] and binomial leap[10] method have been introduced.

C. The chemical master equation

Given an initial condition $\mathbf{x}(t_0)=\mathbf{x}_0$, the probability of being in state $\mathbf{x}$ at time t, $P(\mathbf{x};t)$, satisfies the following discrete PDE:

$$\frac{\partial P(\mathbf{x};t)}{\partial t}=\sum_{j=1}^{M}\alpha_j(\mathbf{x}-\boldsymbol{\nu}_j)P(\mathbf{x}-\boldsymbol{\nu}_j;t)-P(\mathbf{x},t)\sum_{j=1}^{M}\alpha_j(\mathbf{x}). \tag{1}$$

This CME may be written in an equivalent matrix-vector form so that the evolution of the probability density $\mathbf{p}(t)$ [which is a vector of probabilities $P(\mathbf{x};t)$, indexed by the states $\mathbf{x}$] is described by a system of linear, constant coefficient, ordinary differential equations, $\dot{\mathbf{p}}(t)=\mathbf{A}\mathbf{p}(t)$, where the matrix $\mathbf{A}=[a_{ij}]$ is populated by the propensities and represents the *infinitesimal generator* of the Markov process, with $a_{jj}=-\Sigma_{i\neq j}a_{ij}$. Given an initial distribution $\mathbf{p}(0)$, the solution at time t is

$$\mathbf{p}(t)=\exp(t\mathbf{A})\mathbf{p}(0). \tag{2}$$

Recently, Munsky and Khammash[27] made significant progress on the solution of the CME with the finite state projection (FSP) algorithm.

II. THE FSP ALGORITHM

In the FSP algorithm the matrix in Eq. (2) is replaced by $\mathbf{A}_k$, where

$$\mathbf{A}=\left(\begin{array}{c|c}\mathbf{A}_k & * \\ \hline * & *\end{array}\right), \tag{3}$$

i.e., $\mathbf{A}_k$ is a $k\times k$ submatrix of the true operator $\mathbf{A}$. The states indexed by $\{1,\ldots,k\}$ then form the *finite state projection*. The FSP algorithm replaces Eq. (2) with the approximation

$$\mathbf{p}(t_f)\approx\exp(t_f\mathbf{A}_k)\mathbf{p}_k(0), \tag{4}$$

which, by Theorem 2.1 of Ref. 27, is non-negative. The subscript k denotes the truncation just described and we note that a similar truncation is applied to the initial distribution. Consider the column sum $\Gamma_k=\mathbb{1}^T\exp(t_f\mathbf{A}_k)\mathbf{p}_k(0)$, where $\mathbb{1}=(1,\ldots,1)^T$ with appropriate length. Normally, the exact solution (2) would be

```
Algorithm 1: FSP($\mathbf{A},\mathbf{p}(0),t_f,\epsilon$)
  $\Gamma_1:=0$;
  for $k:=1,2,\ldots$ until $\Gamma_k\geqslant 1-\epsilon$ do
      $\Gamma_k:=\mathbb{1}^T\exp(t_f\mathbf{A}_k)\mathbf{p}_k(0)$;
  endfor
  return $\exp(t_f\mathbf{A}_k)\mathbf{p}_k(0)$.
```

A proper probability vector with unit column sum, however, due to the truncation, the sum Γ_k may be less than one, because in the approximate system, probability is no longer conserved. However, as k increases, Γ_k increases too, so that the approximation is gradually improved.[27] Additionally, it is shown in Theorem 2.2 of Ref. 27 that if $\Gamma_k\geqslant 1-\epsilon$ for some prespecified tolerance ϵ, then we have

$$\begin{pmatrix}\exp(t_f\mathbf{A}_k)\mathbf{p}_k(0)\\ \mathbf{0}\end{pmatrix}\leqslant\mathbf{p}(t_f)\leqslant\begin{pmatrix}\exp(t_f\mathbf{A}_k)\mathbf{p}_k(0)\\ \mathbf{0}\end{pmatrix}+\epsilon\mathbb{1}.$$

Algorithm 1 summarizes the FSP. It begins with the matrix representing the CME, $\mathbf{A}$, the initial distribution, $\mathbf{p}(0)$, the time at which the solution to the CME is desired, t_f, and a tolerance, ϵ, specifying how accurate the solution must be. It then gradually increases k in Eq. (4) until the desired level of accuracy is attained. For simplicity, we described the algorithm as if it merely increases k but it can be generalized so that the projection is expanded around the initial state in a way that respects the *reachability*[27] of the model.

A. The Krylov FSP algorithm

The FSP method was recently improved to a Krylov-based approach,[21–23] by adapting Sidje's Expokit codes.[28,29]

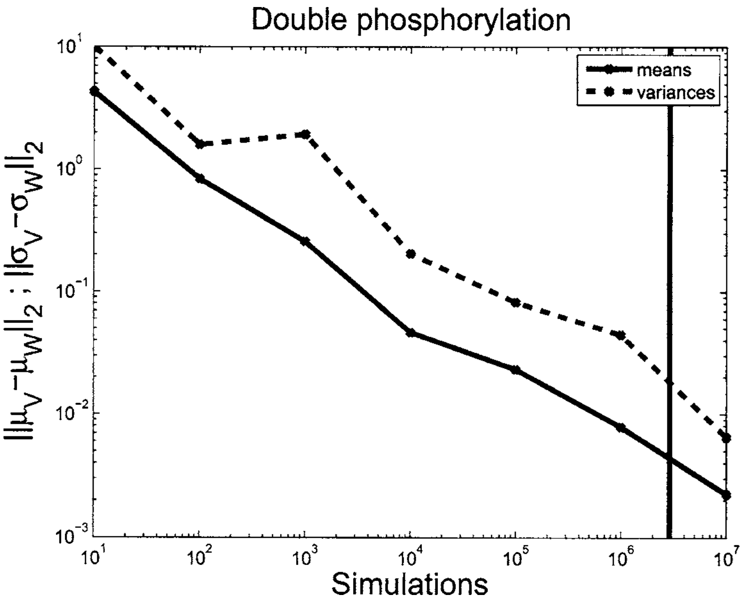

FIG. 1. Application of the CME solver (the Krylov FSP of Sec. II A) and the SSA to the double phosphorylation model in example (e), Table I, Sec. IV B. Estimates of the mean obtained by repeating the SSA, $\boldsymbol{\mu}_V$, or by using the CME solver once, $\boldsymbol{\mu}_W$, are compared in a log-log plot. The vertical line marks the number of simulations that can be performed with the SSA in the same time that it takes to run the CME solver once. Results of a similar comparison for the variances are also plotted.

The Krylov FSP converts the problem of exponentiating a large sparse matrix to that of exponentiating a small, dense matrix in the Krylov subspace. The dimension m of the Krylov subspace is typically small and $m=30$ was used in this implementation. The Krylov approximation to $\exp(\tau\mathbf{A})\mathbf{v}$ is $\beta\mathbf{V}_{m+1}\exp(\tau\bar{\mathbf{H}}_{m+1})\mathbf{e}_1$, where $\beta\equiv\|\mathbf{v}\|_2$, $\mathbf{e}_1$ is the first unit basis vector, and $\mathbf{V}_{m+1}$ and $\bar{\mathbf{H}}_{m+1}$ are the orthonormal basis and upper Hessenberg matrix, respectively, resulting from the well-known Arnoldi process. The exponential in the smaller subspace is computed via the diagonal Padé approximation with degree $p=6$, together with scaling and squaring.

As well as being a matrix-free approach, the Krylov FSP allows the concurrent expansion of the projection and evaluation of the exponential, via the embedded scheme (with vectors padded with zeros to be of consistent sizes as appropriate),

$$\mathbf{p}(t_f) \approx \exp(\tau_K\mathbf{A}_K)\ldots\exp(\tau_0\mathbf{A}_0)\mathbf{p}(0), \quad t_f=\sum_{k=0}^{K}\tau_k, \qquad (5)$$

where the $\{\tau_k\}$ are step sizes and K denotes the total number of steps. Thus, Eq. (5) is evaluated from right to left, harnessing the built-in-step-by-step integration procedure of Expokit.

B. The effectiveness of the CME approach

Trajectorial approaches have been preferred to probability density function (PDF) approaches because a single simulation is relatively cheap. This can be quite a reasonable approach in many applications but for some examples many simulations may be required to accurately approximate the PDF and collect useful statistics, and such an approach does not easily detect when the system settles down to equilibrium. Also, sampling only provides a confidence interval, whereas a PDF approach is accompanied by a certificate of accuracy. Thus, we argue that the two approaches should be viewed as complementary to one another. It is well known that PDF approaches can be computationally demanding but in some cases a CME-based approach can be powerful, as the following two examples demonstrate.

First, Fig. 1 compares the trajectorial and PDF approaches by applying them to the same model of double phosphorylation, which we will return to in example (e) of Sec. IV B. As shown in the figure, the CME approach is more computationally efficient than running many Monte Carlo simulations, for the purposes of computing moments of the distribution. Second, Fig. 1 of Ref. 20 used 50 000 simulations with the SSA to estimate the mean number of molecules of the product species for Michaelis–Menten enzyme kinetics, an example that we will return to in Sec. III F. Applying the Krylov FSP to this same example shows the same trend as in Fig. 1, providing another example for which the CME approach is more computationally efficient, similar to the results of previous studies.[23]

C. The FSP approximation as an example of operator splitting

Despite $\|e^{\mathbf{A}}-e^{\mathbf{A}_k}\|$ being large the FSP approximation performs well for the CME because $\|(e^{\mathbf{A}}-e^{\mathbf{A}_k})\mathbf{p}(0)\|$ is small. Operator splitting and the special structure of the matrices arising in biochemical applications explain why the approximation works so well. For $k=1,2,\ldots$, the sequence of FSP approximations is defined: $\mathbf{A}=\mathbf{A}_k+(\mathbf{A}-\mathbf{A}_k)\equiv\mathbf{A}_k+\mathbf{R}_k$. The matrices that arise in the CME are typically very sparse and the nonzero elements lie in a relatively narrow band around the diagonal. This extra structure is most clearly seen when the state space is ordered by *reachability*, as in Fig. 2, which shows the CME matrix associated with a model of double phosphorylation, described in Sec. IV B.

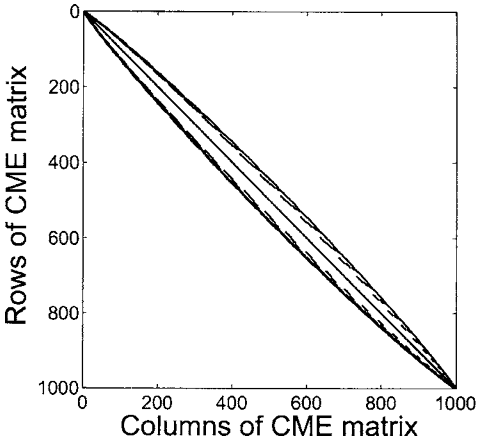

FIG. 2. The CME matrix **A** for double phosphorylation with initial state [10, 10, 0, 0, 0, 0]. 5290 entries are nonzero. The sparse structure is typical of matrices arising in the CME and is important for understanding why the FSP approximation works so well (see Sec. II C).

Thus,

$$\mathbf{A}_k = \left(\begin{array}{cc|c} * & * & \\ & & 0 \\ 0 & * & \\ \hline 0 & & 0 \end{array}\right), \quad \mathbf{R}_k = \left(\begin{array}{cc|c} 0 & & * \\ \hline 0 & * & * \end{array}\right)$$

is a more accurate representation of this structure than Eq. (3), which allows for the general case that the matrix is dense or even full, for example. Under the same ordering the initial distribution is just a unit vector: $\mathbf{p}(0)=[1,0,0,\ldots,0]^T$. From this sparsity pattern, we see that, for sufficiently large k, $\mathbf{R}_k\mathbf{A}_k^i\mathbf{p}(0)$ is zero, and as we increase k it remains zero for larger i. Using the 10×10 principal submatrix in the top left hand corner of the full matrix in Fig. 2 as an example, $\mathbf{R}_{10}\mathbf{A}_{10}^i\mathbf{p}(0)=0$, for $i=0,1,2$ but $\mathbf{R}_{10}\mathbf{A}_{10}^3\mathbf{p}(0)\neq 0$. If we increase the projection size k, $\mathbf{R}_{20}\mathbf{A}_{20}^i\mathbf{p}(0)$ is zero for $i=0,1,\ldots,4$, but nonzero for higher powers. For sufficiently large k, $\mathbf{R}_k\mathbf{A}_k^i\mathbf{p}(0)$ is zero for $i=1,\ldots,j$ and the following series for the error can be derived:

$$(e^{\mathbf{A}}-e^{\mathbf{A}_k})\mathbf{p}(0)=\sum_{n=1}^{\infty}\frac{1}{n!}((\mathbf{A}_k+\mathbf{R}_k)^n\mathbf{p}(0)-\mathbf{A}_k^n\mathbf{p}(0))$$

$$=\sum_{n=j+2}^{\infty}\frac{1}{n!}\left(\sum_{i=0}^{n-(j+2)}\mathbf{A}^i\mathbf{R}_k\mathbf{A}_k^{n-i-1}\mathbf{p}(0)\right).$$

In particular, the first $j+1$ terms are zero, and, as we increase the size of the projection, more and more of the leading terms become zero. This explains how the extra structure in the matrices representing the CME make them very suitable to the FSP approximation.

Similar arguments may be used to understand the approximation from the perspective of the Baker–Campbell–Hausdorff (BCH) formula.[30] Suppose k is sufficiently large such that $\mathbf{R}_k\mathbf{p}(0)=0$ so $e^{\mathbf{R}_k}\mathbf{p}(0)=\mathbf{p}(0)$. By application of the BCH formula with $\mathbf{A}=\mathbf{A}_k+\mathbf{R}_k$,

$$e^{\mathbf{A}_k}\mathbf{p}(0)=e^{\mathbf{A}_k}e^{\mathbf{R}_k}\mathbf{p}(0)=e^{\mathbf{A}+(1/2)[\mathbf{A}_k,\mathbf{R}_k]+\cdots}\mathbf{p}(0).$$

The FSP approximation appears on the left and on the right we see an expression involving the original matrix $\mathbf{A}$ and terms involving the commutator $[\mathbf{A}_k,\mathbf{R}_k]:=\mathbf{A}_k\mathbf{R}_k-\mathbf{R}_k\mathbf{A}_k$ and higher order commutators. The approximation is exact if the split operators commute but otherwise the magnitude of the error is governed by the magnitude of the commutators. Empirically, $\|[\mathbf{A}_k,\mathbf{R}_k]\|$ is often quite large but if we also apply truncation to the commutator we find that $[\mathbf{A}_k,\mathbf{R}_k]_k=0$. In fact, for sufficiently large k, $[\mathbf{A}_k,\mathbf{R}_k]\mathbf{p}(0)=0$ and, similar to the above result, more and more of the higher order terms in the BCH formula are seen to become zero as the projection size increases.

As an example of operator splitting, the FSP approximation is unusual because it discards the effect of one component, namely, $\mathbf{R}_k$. In the next section we consider a more conventional example of operator splitting.

III. APPLICATION OF THE QSSA TO THE CME

Motivated by the successful application of the QSSA to the SSA,[13–15,18–20] we now apply the QSSA in the context of the CME. Previous works[12,31] have considered related ideas but the methods presented here are based on Krylov methods combined with aggregation.

A. Operator splitting in the CME

We begin with the same partition of the reactions into fast and slow subsets and the same induced decomposition of the state space into "virtual fast processes," that is used by the slow-scale SSA (ssSSA) (Ref. 13) and the nested SSA (nSSA),[14,15] and proceed to place these approximations in a matrix framework. As in Sec. III of Ref. 13 the fast reactions induce a "fast partition" of the state space, with two states being in the same subset of this partition if and only if one can be reached from the other via a sequence of fast reactions. Each subset gives rise to a "virtual fast process," defined in Sec. IV of Ref. 13, which consists of the subsystem obtained when the slow reactions are turned off.

Let $R_f\subset\{R_1,\ldots,R_M\}$ denote the subset of fast reactions and let R_s denote the rest. The CME [Eq. (1)] can then be rewritten by splitting the right hand side into two parts. The fast reactions give rise to the following "fast CME,"

$$\frac{\partial P_f(\mathbf{x};t)}{\partial t}=\sum_{j\in R_f}\alpha_j(\mathbf{x}-\boldsymbol{\nu}_j)P_f(\mathbf{x}-\boldsymbol{\nu}_j;t)-P_f(\mathbf{x};t)\sum_{j\in R_f}\alpha_j(\mathbf{x}). \qquad (6)$$

An analogous "slow CME" arises for R_s and summing the two recovers Eq. (1). The same splitting may be expressed conveniently in matrix notation as

$$\mathbf{A}=\mathbf{A}_f+(\mathbf{A}-\mathbf{A}_f)\equiv\mathbf{A}_f+\mathbf{A}_s.$$

Here $\mathbf{A}_f$ corresponds to the fast CME so that, in matrix notation, Eq. (6) is $\dot{\mathbf{p}}_f(t)=\mathbf{A}_f\mathbf{p}_f(t)$. Similarly, $\mathbf{A}_s$ corresponds to the slow CME, which in matrix notation is $\dot{\mathbf{p}}_s(t)=\mathbf{A}_s\mathbf{p}_s(t)$. Both $\mathbf{A}_f$ and $\mathbf{A}_s$ are infinitesimal generators of Markov processes by themselves, a property deliberately preserved in order for them to be amenable to further analysis.[13]

For many important biological examples, the matrix $\mathbf{A}_f$ is block diagonal, with blocks corresponding to subsets of the fast partition and each block being much smaller than the original matrix. Thus, each block governs a virtual fast process with its own stationary distribution. It is these distributions that are used by the nSSA and ssSSA to approximate the modified propensities of the slow reactions, the so-called "slow-scale propensity" functions. Thus, in general, $\mathbf{A}_f$ has multiple zero eigenvalues, corresponding to distinct eigenvectors, and similar remarks apply to $\mathbf{A}_s$.

B. A splitting scheme based on the QSSA

We expect the combined cost of independently exponentiating $\mathbf{A}_s$ and $\mathbf{A}_f$ to be less than that of treating the full system because of the block diagonal structures of $\mathbf{A}_s$ and $\mathbf{A}_f$. Thus, we consider taking a small time step h with a splitting scheme,

$$\mathbf{p}(t+h)=e^{h\mathbf{A}}\mathbf{p}(t)\approx e^{(1/2)h\mathbf{A}_f}e^{h\mathbf{A}_s}e^{(1/2)h\mathbf{A}_f}\mathbf{p}(t).$$

This is the distribution that must be approximated and then sampled from in order to take a small step in a simulation algorithm such as the nSSA or ssSSA. The approximation being used is the symmetric Strang splitting,[32] which is of order 2.

When using the QSSA, the time step h is required to be sufficiently large such that the fast reactions almost reach equilibrium, so that $e^{(1/2)h\mathbf{A}_f}$ is approximated by its stationary solution, $\mathbf{A}_{f_\infty}\equiv\lim_{t\to\infty}e^{t\mathbf{A}_f}$. Introducing this approximation to the Strang splitting gives the approximation

$$\mathbf{p}(t+h)\approx\mathbf{A}_{f_\infty}e^{h\mathbf{A}_s}\mathbf{A}_{f_\infty}\mathbf{p}(t). \tag{7}$$

Note that the action of $\mathbf{A}_{f\infty}$ is the analog of (14) in step 3 of the slow-scale algorithm[13] and that the two constraints on the size of the time step h are analogous to those made by the slow-scale approximation lemma[13] that requires h be small enough that only a single slow reaction occurs over the interval but still large compared to the relaxation time of the fast reactions. Thus, the approximation is a good analog of the ssSSA. Also, note that $\mathbf{A}_{f_\infty}$ is a projection matrix so $\mathbf{A}_{f_\infty}^2=\mathbf{A}_{f_\infty}$, which could be used for computational savings when approximating $\mathbf{p}(Nh)$ by taking N steps with Eq. (7). We now introduce the approximation $\mathbf{A}_{f_\infty}e^{h\mathbf{A}_s}\mathbf{A}_{f_\infty}\approx e^{h\hat{\mathbf{A}}_s}$ into Eq. (7) giving our penultimate approximation to Eq. (2),

$$e^{t\hat{\mathbf{A}}_s}\mathbf{p}(0). \tag{8}$$

In analogy with the Krylov approximation, where the projection of the exponential is approximated by the exponential of the projection,[33] we consider choosing $\mathbf{A}_{f_\infty}\mathbf{A}_s\mathbf{A}_{f_\infty}$ for $\hat{\mathbf{A}}_s$. However, this may not be Markovian so we adopt the ansatz that $\hat{\mathbf{A}}_s\equiv\mathbf{A}_{f_\infty}\mathbf{A}_s\mathbf{A}_{f_\infty}+\alpha(\mathbf{A}_{f_\infty}-I)$, which is Markovian for suitably large α,[34] for example $\alpha\geqslant\max_i|a_{ii}|$. In summary, we have approximated one Markov process, governed by the CME represented by $\mathbf{A}$, with another Markov process, governed by the CME represented by $\hat{\mathbf{A}}_s$.

C. The QSSA as a form of aggregation

Briefly, we introduce the aggregation and disaggregation operators, $\mathbf{E}$ and $\mathbf{F}$.[34,35] Given the state space, of size n_A and some partition of this into n_B subsets, we define $\mathbf{E}\in\mathbb{R}^{n_B\times n_A}$ such that $\mathbf{E}_{i,j}=1$, if state j is in subset i and $\mathbf{E}_{i,j}=0$ otherwise. We are then free to choose any $\mathbf{F}\in\mathbb{R}^{n_A\times n_B}$ with non-negative entries, unit column sum, and such that $\mathbf{F}_{i,j}^T\neq 0$ if and only if $\mathbf{E}_{i,j}\neq 0$. Usually, we think of $n_B\ll n_A$. The pair of operators always have the properties that $\mathbf{EF}=\mathbf{I}$, $\mathbf{FE}$ is a projection matrix, and $\mathbf{EAF}$ also represents a Markov process whenever $\mathbf{A}$ does. The technique of aggregation was introduced so that the former could be used as an approximation to the latter, with the dual computational advantages of reducing the dimension (a matrix of dimension n_B as opposed to n_A) while still preserving the Markov property.

We choose $\mathbf{E}$ to combine states according to the partition of the state space into virtual fast processes and we choose $\mathbf{F}$ so that its columns record the equilibrium solutions of these fast processes. With this choice,

$$\mathbf{A}_{f_\infty}=\mathbf{FE}. \tag{9}$$

Thus,

$$\mathbf{E}\hat{\mathbf{A}}_s=\mathbf{E}(\mathbf{A}_{f_\infty}\mathbf{A}_s\mathbf{A}_{f_\infty}+\alpha(\mathbf{A}_{f,\infty}-I))=(\mathbf{EF})(\mathbf{EA}_s\mathbf{F})\mathbf{E}+0=\mathbf{BE},$$

where we have introduced $\mathbf{B}\equiv\mathbf{EA}_s\mathbf{F}$. (In fact, $\mathbf{EA}_f=0$ so $\mathbf{B}=\mathbf{EAF}$, which is the conventional approximation used when the technique of aggregation is applied.) Equivalently,

$$\mathbf{E}e^{t\hat{\mathbf{A}}_s}=e^{t\mathbf{B}}\mathbf{E}. \tag{10}$$

Importantly, this explicitly gives a more efficient way to compute Eq. (8). We recover disaggregated solutions as $\mathbf{F}e^{t\mathbf{B}}\mathbf{E}\mathbf{p}(0)$, which is our final approximation to Eq. (2). This is mathematically equivalent to approximating Eq. (8) by $\mathbf{A}_{f_\infty}e^{t\hat{\mathbf{A}}_s}\mathbf{p}(0)$ but computationally preferable. The Markov model governed by $\mathbf{B}$ may be thought of as being obtained from $\mathbf{A}$ by combining states in each virtual fast process into one big super state. The propensities that populate $\mathbf{B}$ correspond to moving between these super states. Each propensity is the same as the slow-scale propensity function used by the ssSSA, which is the weighted average of the regular propensities over the states in the virtual fast process, treated as though they were in their equilibrium distribution. Thus, we have achieved our goal of placing the application of the QSSA to the CME, along with the approximations of the nSSA and ssSSA, within the framework of aggregation.

D. Computation of quasistationary distributions

We outline four strategies for obtaining the stationary solutions of $\mathbf{A}_f$, which we need to define the action of $\mathbf{F}$. The first approach is described in Appendix A of Ref. 13, where a recursive formulation for the stationary solution may be derived by making the ansatz that the model satisfies the special criterion of *detailed balance*.[3] We identify a formula in this way for the Michaelis–Menten model. Secondly, a Monte Carlo strategy would be to repeatedly simulate each virtual fast process and estimate the stationary solution. A variation of this would be to use Eq. (13) of Ref. 14, as in the nSSA, to directly estimate the propensities that populate the matrix $\mathbf{B}$. We suggest identification of the blocks, $\mathbf{A}_{f_i}$, of the fast operator, via the reachability structure of the model, and then to use off-the-shelf methods for solving the matrix equation $\mathbf{A}_{f_i}\mathbf{x}_i=0$. Two good choices would be LU factorization (for relatively small, dense blocks) or the power method (for larger, sparse blocks) and the implementation for this paper used a combination of these. In order to overcome the singular nature of the blocks one can use the trick of adding a rank one matrix[36] by applying the LU solver to $(\mathbf{A}_{f_i}+\alpha e_j e_j^T)\tilde{\mathbf{x}}_i=e_j$ and normalize the result to obtain $\mathbf{x}_i$. By consideration of the Gerschgorin disks, the choice of $\alpha \equiv 2\max|a_{jj}|$ ensures that the use of the power method with the shifted operator $(\mathbf{A}_{\mathbf{f}_i}+\alpha I)$ converges to the correct eigenvalue. A good choice for the initial vector would be $[1,\ldots,1]^T$ because it is guaranteed to have a nonzero projection onto the stationary solution and is orthogonal to all other eigenvectors.

E. The QSSA-based CME solver

The QSSA-based CME solver that evaluates Eq. (10) is outlined in Algorithm 2. It begins with the matrix representing the CME, $\mathbf{A}$, the initial distribution, $\mathbf{p}(0)$, the time at which the solution to the CME is desired, t_f, the set of fast reactions, R_f, and a tolerance, ϵ, that will be used later in a call to the FSP algorithm. The preprocessing stage uses the fast reactions R_f to compute $\mathbf{A}_f$ from $\mathbf{A}$, and also to compute $\mathbf{E}$, which represents the partition of the state space into virtual fast processes. Next, $\mathbf{A}_f$ is used to compute $\mathbf{F}$ via any of the techniques in Sec. III D. Next, $\mathbf{B}$ is formed, by computing $\mathbf{B}=\mathbf{EAF}$. Next, the FSP (Algorithm 1) is used to compute $e^{t_f\mathbf{B}}\mathbf{Ep}(0)$. We employ a modification of the FSP that uses Krylov techniques, described in Sec. II A. Finally, there is a post processing step, which is equivalent to multiplication of the resulting distribution by $\mathbf{F}$. If the approximation were exact we would have $\mathbf{Fq}(t_f)=\mathbf{p}(t_f)$.

```
**Algorithm 2:** QSSA CME solver $(\mathbf{A},\mathbf{p}(0),t_f,\epsilon,R_f)$
    $[\mathbf{E},\mathbf{F},\mathbf{A}_f]$=Preprocess$(\mathbf{A},R_f)$;
    $\mathbf{B}=\mathbf{EAF}$;
    $[\mathbf{q}(t_f)]$=FSP $(\mathbf{B},\mathbf{Ep}(0),t_f,\epsilon)$;
    return $\mathbf{Fq}(t_f)$.
```

Note that the remarks made in Refs. 13 and 30 carry over to the QSSA-based CME solver. First, in many cases it is only the aggregated distributions that are of interest so computational savings may be made by skipping the postprocessing step. Second, the computation of the stationary solutions is trivially parallelized and can be automated as it can be for related methods such as the ssSSA. Often, it is only an approximation to the first few moments that is required, allowing savings in computations with $\mathbf{F}$. Also, by treating each virtual fast process separately $\mathbf{B}$ can be computed using only parts of $\mathbf{E}$ and $\mathbf{F}$ at any one time so that we never need store these matrices in full. Third, we compare the aggregated distributions as a measure of the accuracy of the approximation. For example, in the case of the Michaelis–Menten enzyme kinetics, the accuracy is assessed in terms of the distribution of products, and more generally the accuracy is assessed as $\|\mathbf{E}e^{t_f\mathbf{A}}\mathbf{p}(0)-e^{t_f\mathbf{B}}\mathbf{Ep}(0)\|$. Fourth, experimental data are often so difficult to obtain that there is only enough information to parameterize an aggregated model such as the QSSA.

TABLE I. Michaelis–Menten scheme, as in Ref. 20. Rate constants: $c=[1.0,1.0,0.1]$. Initial state: $[100,\ E_I,\ 0,\ 0]$. Example (i): $E_I=10$, $t_f=30$. Example (ii): $E_I=1000$, $t_f=20$.

	Reaction	Propensity
1	$S+E\rightarrow C$	$c_1\times S\times E$
2	$S+E\leftarrow C$	$c_2\times C$
3	$C\rightarrow P+E$	$c_3\times C$

F. The tQSSA-based CME solver

Having discussed the interpretation of the QSSA in the stochastic setting, and, in particular, its application to the CME, we now establish the connection to the total QSSA in the deterministic setting. The original papers detailing the tQSSA in the ODE setting[37–41] introduce it via the example of Michaelis–Menten kinetics, while Rao and Arkin use the same example to introduce their QSSA in the stochastic setting.[20] The connection to the deterministic literature on the tQSSA shows that Algorithm 2 is really a natural generalization of the tQSSA to the stochastic setting so from now on we refer to Algorithm 2 as the tQSSA CME solver.

The Michaelis–Menten model involves an enzyme, E, that gradually catalyzes the conversion of all available substrate S, into a product P, via an intermediate complex C. There are four chemical species $[S,E,C,P]$, and three chemical reactions, which are described in Table I. The system is subjected to the following pair of conservation laws: $E_I=E+C$ and $S_I=S+C+P$. Reactions 1 and 2 are "fast." Figure 3(a) shows the full state space of the Michaelis–Menten scheme and Fig. 3(b) shows the partition into virtual fast processes. In relation to Algorithm 2, $\mathbf{E}$ encodes the partition into the rectangles of the figure, $\mathbf{F}$ encodes the stationary distribution of the process in each rectangle, and the slow-scale propensities that populate $\mathbf{B}$ correspond to transitions from one rectangle to the next rectangle higher up.

Taking into account the conservation laws, the deterministic, reaction rate equation model for the Michaelis–Menten system consists of two ODEs: one each for the substrate and the complex. It is usual to make the approximation $dC/dt=0$ and then reduce the system to just one ODE, which is known as the standard QSSA (sQSSA). The tQSSA makes the same approximation after first introducing the change of variable known as the *total substrate*, $S_T\equiv S+C$, which greatly extends the parameter regime over which the approximation is valid.[41] This motivates us to consider another partition of the state space, in the stochastic setting, obtained by combining states with the same value of S_T. In fact, as the figure shows, such a partition is the same as the partition into virtual fast processes. Thus, Algorithm 2 may be regarded as a generalization of the tQSSA to the stochastic setting and if the closely related approximations of the nSSA (Ref. 14) and ssSSA (Ref. 13) were applied to the Michaelis–Menten scheme, they would also be classified as examples of the tQSSA. For example, the total substrate is the same as the invariant variable[14,15] used in the analysis of the nSSA, which reflects the conservation laws of the virtual fast processes.

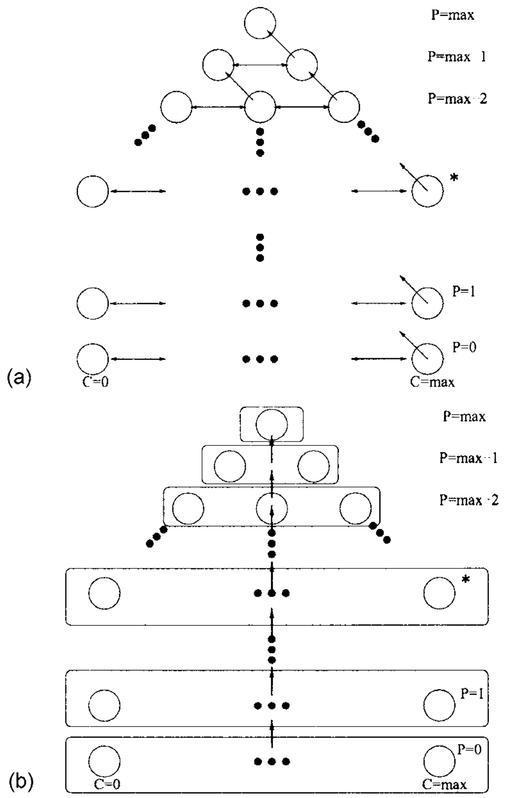

FIG. 3. (a) Each circle represents a state $[S,E,C,P]$, in the Michaelis–Menten model. The initial state, $[S_I,E_I,0,0]$, is the bottom left circle. The top circle is $[0,E_I,0,S_I]$, an absorbing state reached when all substrates have been converted to products, so $P=P_{\max}=S_I$. Transitions within the same row are the fast, reversible formation (to the right) and dissociation (to the left) of the complex. Upward transitions between rows represent formation of product. (b) The rectangles partition the state space into virtual fast processes. States within the same rectangle have the same value of the total substrate, $S_T\equiv S+C$.

The Appendix provides more details, including a formula for **B** and the relationship to the sQSSA.

IV. RESULTS

We compare the accuracy and efficiency of (A) the Krylov FSP for the full CME and (B) the tQSSA-based CME A solver. By default, the Krylov FSP is called with (Expokit, FSP) tolerances of $(10^{-8},10^{-5})$, but bear in mind that these bounds are pessimistic and the actual results may be better. The postprocessing step is not included in the runtimes reported here. Unless otherwise stated, all numerical experiments used FORTRAN with the Intel "ifort" compiler, and were conducted on an SGI Altix with 64 Itanium 2 CPUs and 120 GBytes of memory running the LINUX operating system. However, only a single processor was used. Since the true solution is not available, we assess the accuracy of the tQSSA by comparison with the Krylov FSP, with strict tolerances.

TABLE II. Comparison of Krylov FSP and tQSSA for the Michaelis–Menten examples in Table I. The speed-up is defined as the runtime of the Krylov FSP divided by the runtime of the Krylov FSP divided by the runtime of the tQSSA.

Example	Speed-up	$\|\cdot\|_1$	$\|\cdot\|_2$	$\|\cdot\|_\infty$
(i)	1	7E−3	2E−3	4E−4
(ii)	216	3E−4	9E−5	3E−5

We compare the approximations of the mean obtained by the two methods. The mean refers to the average number of molecules of a chemical species at t_f. The mean of a distribution involving more than one species refers to the vector of means for each species: $[\mathbb{E}(S_1),\ldots,\mathbb{E}(S_N)]$.

A. Michaelis–Menten enzyme kinetics

The results of applying the tQSSA-based CME solver to the two examples in Table II, are recorded in Table III, which shows that it is an extremely good approximation. The accuracy is measured by comparing the conditional distributions for the products. Example (ii) shows considerable savings in runtime, while example (i) is really too small to see this. The tQSSA is more accurate for example (ii) where the enzymes are in excess, which is to be expected since the increased population of enzymes increases the propensity of the fast reactions making the assumptions underlying the tQSSA even more appropriate. In particular, we do not encounter the trouble that Rao and Arkin report for this example, which is an advantage of the tQSSA approach.

The norms of the operators involved in the examples in Table II are given in Table III. The norm of the reduced operator **B** is at least two orders of magnitude less than that of the full model **A** and this is where some of the computational savings are being made. Also, $\|[\mathbf{A}_s,\mathbf{A}_f]\|_2/\|\mathbf{A}\|_2$ is small, which is consistent with analysis via the BCH formula. Further experiments show that $\|[\mathbf{A}_s,\mathbf{A}_f]\mathbf{p}(t)\|_2$ becomes much smaller for larger t. This suggests using the full Krylov FSP for a brief initial transient, and then switching to the tQSSA for the rest of the computation, would increase the accuracy of the tQSSA without much extra cost. Numerical experiments combining the algorithms confirm this for example (ii): using the full Krylov FSP for an initial transient of $t=1.0$ and then switching to the tQSSA for the rest of the integration only increases the runtime to about 10 s but gives significantly greater accuracy of 10^{-6} in the 1-norm.

B. Double phosphorylation

This is an example of a fully competitive reaction scheme, with substrates competing for a common enzyme, and arises in the double phosphorylation of MAPK by MAPKK.[38,42] It can be thought of as one Michaelis–Menten

TABLE III. Norms of operators in Michaelis–Menten model. $\mathbf{C}\equiv[\mathbf{A}_s,\mathbf{A}_f]$.

Ex	$\|\mathbf{A}\|_2$	$\|\mathbf{A}_f\|_2$	$\|\mathbf{A}_s\|_2$	$\|\mathbf{C}\|_2$	$\|\mathbf{Ap}(0)\|_2$	$\|\mathbf{Cp}(0)\|_2$	$\|\mathbf{B}\|_2$
(i)	1.7E3	1.7E3	1.7	1.7E2	1.4E2	1.4E2	1.97
(ii)	1.9E5	1.9E5	19.1	1.8E4	1.4E5	1.4E4	19.1

TABLE IV. Description of the double phosphorylation enzyme kinetics scheme. The initial state is $[100, E_I, 0, 0, 0, 0]$. Examples (a), (b), and (c) use $E_I=1000,100,10$, respectively, and $t_f=2,2.5,20$, respectively. Examples (a), (b), and (c) use $c=[0.2,1.0,0.6,0.2,1.0,0.5]$ (Ref. 40). Examples (d), (e), and (f) match (a), (b), and (c), respectively, except that they have different rate constants: $c=[1.0,1.0,0.1,1.0,1.0,0.1]$.

	Reaction	Propensity
1	$S_1+E\rightarrow C_1$	$c_1\times S_1\times E$
2	$S_1+E\leftarrow C_1$	$c_2\times c_1$
3	$C_1\rightarrow S_2+E$	$c_3\times C_1$
4	$S_2+E\rightarrow C_2$	$c_4\times S_2\times E$
5	$S_2+E\leftarrow C_2$	$c_5\times C_2$
6	$C_2\rightarrow P+E$	$c_6\times C_2$

scheme feeding into another and so there is a natural choice for the tQSSA in which two new "total substrate" variables are introduced: $S_{T_i}\equiv S_i+C_i$ for $i=1,2$. There are six chemical species $[S_1,E,C_1,S_2,C_2,P]$, and six chemical reactions, described in Table IV, which gives parameters for six examples considered here. It is subject to two conservation laws: $S_{1_I}=S_1+C_1+S_2+C_2+P$ and $E_I=E+C_1+C_2$ and has an absorbing state that is always eventually reached. The pair of reversible reactions 1 and 2, and the pair 4 and 5, are deemed to be fast and the remainder slow.

Following Pedersen *et al.*,[38] together with the two conservation laws, the deterministic model is described by the following four coupled ODEs:

$$\frac{dS_1}{dt}=c_2C_1-c_1S_1E,$$

$$\frac{dC_1}{dt}=c_1S_1E-(c_2+c_3)C_1,$$

$$\frac{dS_2}{dt}=c_3C_1+c_5C_2-c_4S_2E,$$

$$\frac{dC_2}{dt}=c_4S_2E-(c_5+c_6)C_2. \qquad (11)$$

The sQSSA then makes the approximation $dC_i/dt\approx 0$, for $i=1,2$, while the tQSSA makes the same approximation after first introducing the change of variables mentioned above. Either approach reduces the system to just two DEs, although they are parameterized by more complicated propensity functions. In particular, the tQSSA gives rise to a cubic polynomial that must be solved at each step of a numerical integration scheme. Figure 4(a) shows the tQSSA is very accurate in the deterministic setting for example (c). Using MATLAB's built-in ODE solvers for example (b) shows that the stiff solver *ode15s* is slightly faster than *ode45* and that using Cardan's formula, for the solution of a cubic, is twice as fast as *fzero*.

Figure 4(b)compares a stochastic trajectory with the deterministic model. The dynamics are roughly consistent, although the stochastic trajectory is absorbed at $t\approx 13$, while at that time the deterministic model [Eq. (11)] is just reaching equilibrium. ODE models for chemical kinetics are in terms of concentrations (in units of moles per liter) and the rate constants are closely related to, but not quite the same as, the c_i used by the SSA.[4] For numerical testing, we use the same values of c_i for both models but for other applications appropriate scalings would need to be taken into account.

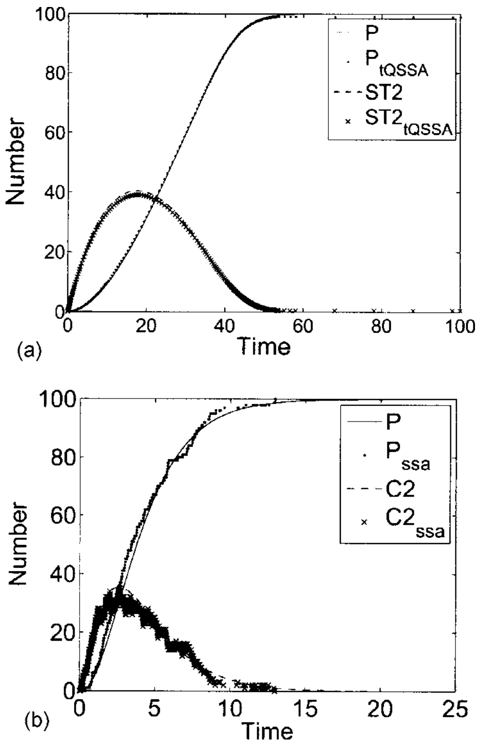

FIG. 4. (a) Results of using Eq. (11) and the tQSSA for the deterministic model of double phosphorylation for example (c) in Table IV are superimposed for comparison. (b) Results of using the stochastic and deterministic models of double phosphorylation for example (b) in Table IV are superimposed for comparison.

Table V compares the tQSSA with the full Krylov FSP. For each example, we choose a value of t_f that occurs at an "interesting" stage of the dynamics of the process, roughly just before the peak in the population of the second complex, C_2, and still far from equilibrium. The results are significant but it is anticipated that choosing larger values of t_f would favor the tQSSA even more. Examples (a), (b), and (c) show the effect of reducing the number of enzymes. For examples (a) and (b), the Krylov FSP eventually uses a matrix of size 4 517 885, which is about 98% of the full model (of size 4 598 126). For both examples, almost all of the computational time of the tQSSA is spent preprocessing, with less than one second being needed to solve the reduced system. Some savings in preprocessing can be made by applying Algorithm 2 to a truncated version of the operator. For the double phosphorylation example, we use the same truncation size as the Krylov FSP uses but other ways to choose the truncation size will be considered in future work.

The enzymes are well in excess for example (a), for which the speed-up is more than an order of magnitude while maintaining reasonable accuracy. Visualizations of the solution are provided in Fig. 5, which shows that the tQSSA can be very effective. The enzymes and substrates are balanced in example (b), for which the tQSSA shows a speed-up of about a factor of 4. This is less than the last example, as is

TABLE V. Comparison of Krylov FSP (**A**) and tQSSA (**B**) for the double phosphorylation model, with examples as in Table IV. The accuracy of the tQSSA is assessed in terms of the conditional distribution for the products P, the mean of which is recorded in the last column. For each method, n is the size of the projection used.

Example		Runtime (s)	$\lVert\cdot\rVert_1$	$\lVert\cdot\rVert_2$	$\lVert\cdot\rVert_\infty$	n	$\mathbb{E}(P)$
(a)	**A**	7.446				4,517,885	29.4
	B	356	3E−2	8E−3	3E−3	5,151	29.7
(b)	**A**	1,414				4,517,885	28.2
	B	353	0.6	0.1	4E−2	5,151	31.8
(c)	**A**	60				270,272	31.4
	B	5	0.3	5E−2	1E−2	5,007	33.3
(d)	**A**	7,567				1,782,721	1.745
	B	144	2E−3	1E−3	6E−4	5,151	1.749
(e)	**A**	1,227				1,869,423	2.1
	B	151	9E−2	4E−2	2E−2	5,007	2.2
(f)	**A**	5				32,967	1.6
	B	1.2	6E−2	3E−2	2E−2	5,007	1.7

the accuracy. A comparison of the solutions appears very similar to that of Fig. 5, although the tQSSA has "shifted" the distribution slightly to the right, which is consistent with our intuition that the tQSSA overestimates how quickly the reactions progress towards equilibrium. The enzymes have been reduced so much in example (c) that the substrates are now in excess and the system is truly competitive. Compared to the previous examples, we expect that the much larger value of t_f would be favorable towards the tQSSA, while the reduced number of enzymes would be unfavorable. Overall, the accuracy is intermediate between the first and second examples, while the speed-up is about an order of magnitude. The CME solutions obtained from the two methods show a similar comparison as in example (b).

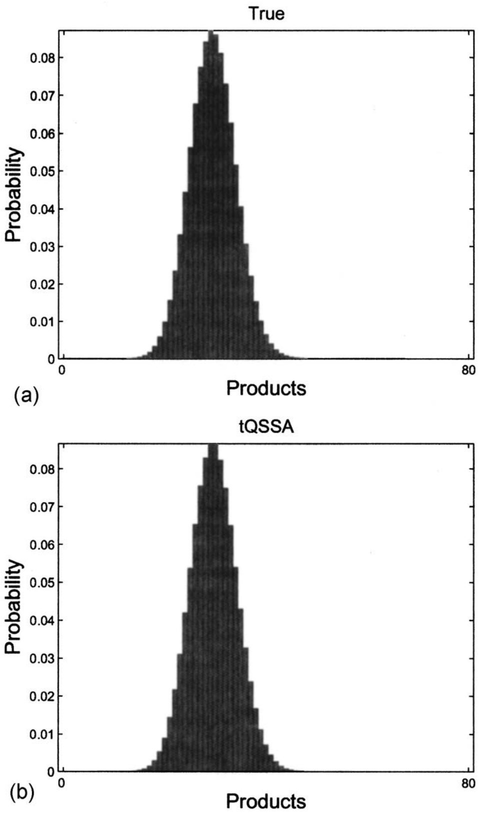

FIG. 5. The CME solution for example (a) in Table IV. (a) The (true) CME solution computed with the full Krylov FSP. (b) Result of using the tQSSA.

Examples (d), (e), and (f) show the effects of changing the rate constants to be in line with a coupled pair of Michaelis–Menten schemes taken from Ref. 20. This change makes the problem even more suitable to the tQSSA as the difference between the propensities for fast and slow reactions is more pronounced. Thus, examples (d), (e), and (f) show better accuracy than examples (a), (b), and (c), respectively, and a glance at $\mathbb{E}(P)$ shows the systems have converged much more quickly towards equilibrium. Compared to (a) and (b) the projection size is significantly reduced in (d) and (e), respectively, but this is not reflected in the runtimes because the new rate constants force Expokit to use smaller step sizes. This stiffness is overcome through aggregation and the tQSSA for these examples is about twice as fast as the full Krylov FSP.

C. Goldbeter–Koshland switch

The Goldbeter–Koshland switch[43] consists of a pair of Michaelis–Menten enzyme kinetic models, catalyzed by different enzymes, in which the product of the one forms the substrate of the other, and vice versa. There are six chemical species $[S, E_1, C_1, P, E_2, C_2]$, and six chemical reactions, which are described in Table VI. It is subject to three conservation laws: $S_I = S + C_1 + P + C_2$, $E_{1_I} = E_1 + C_1$, and $E_{2_I} = E_2 + C_2$. Reactions 1, 2, 4, and 5 are fast. The full model uses a projection of size $\approx 170\,000$ but the tQSSA drastically reduces this to approximately 100. This gives a speed-up of a factor of 5, while maintaining accuracy of 10^{-3}, for the distribution of S_{T_1}. About 90% of the runtime of the tQSSA is spent preprocessing. Again, the means compare well: $\mathbb{E}(S_{T_1})$

TABLE VI. The Goldbeter–Koshland switch (Ref. 43). We set the initial state to [100, 100, 0, 0, 100, 0], $c=[1.0,1.0,0.1,1.0,1.0,0.1]$, and $t_f=20$.

	Reaction	Propensity
1	$S+E_1 \rightarrow C_1$	$c_1 \times S \times E_1$
2	$S+E_1 \leftarrow C_1$	$c_2 \times C_1$
3	$C_1 \rightarrow P+E_1$	$c_3 \times C_1$
4	$P+E_2 \rightarrow C_2$	$c_4 \times P \times E_2$
5	$P+E_2 \leftarrow C_2$	$c_5 \times C_2$
6	$C_2 \rightarrow S+E_2$	$c_6 \times C_2$

and $\mathbb{E}(S_{T_2})$ are 51.10 and 48.91, respectively, under **A**, while they are 51.09 and 48.91, respectively, under **B**.

D. The mitogen activated protein kinase cascade

Recently, the deterministic tQSSA was applied to coupled enzymatic networks similar to those in this paper, including the Goldbeter–Koshland switch.[44] The authors expressed the desire to generalize the deterministic tQSSA to the stochastic framework of the CME. We provide such a generalization. As an example we apply the tQSSA to the full MAPK cascade,[42] a large coupled enzymatic network implicated in a variety of signaling processes governing transitions in a cell's phenotype. Figure 6 is a schematic representation of the reactions in the MAPK cascade, and a complete description of the reactions can be found in Ref. 42.

The MAPK cascade has a modular structure and is composed of ten coupled Michaelis–Menten schemes, consisting of 22 species and 30 reactions in all. We use rate constants for the Michaelis–Menten building blocks that match the Michaelis–Menten model already studied in Table I. Thus, there are ten slow reactions with rate constant of 0.1 and the rest are fast reactions with rate constant of 1.0. For the initial state, we use 100 molecules each for E_1, E_2, $KKP'ase$, $KP'ase$, KKK, KK, and K (these are the seven key species that form the nonzero elements of the natural initial condition used by Huang and Ferrel[42]) and all other species are set to zero.

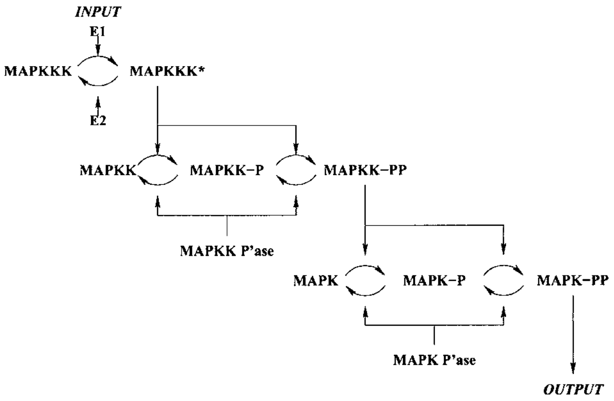

FIG. 6. Schematic of the MAPK cascade, adapted from Ref. 42. It involves a MAPK kinase kinase (MAPKKK), a MAPK kinase (MAPKK), and a MAPK. We follow standard notation (Ref. 42) and abbreviate MAPKKK to KKK, MAPKK to KK, and MAPK to K. KKK* denotes activated MAPK. K-P and K-PP denote singly and doubly phosphorylated MAPK, respectively. P'ase denotes phosphotase.

The structure of the MAPK cascade gives rise to the following eight total substrates:

$$\begin{aligned}
S_{T_1} &\equiv KKK + KKK\,.\,E_1,\\
S_{T_2} &\equiv KKK^* + KKK^*\,.\,E_2 + KKK^*\,.\,KK\\
&\quad + KKK^*\,.\,KK-P,\\
S_{T_3} &\equiv KK + KKK^*\,.\,KK,\\
S_{T_4} &\equiv KK-P + KK-P\,.\,KKP'ase + KKK^*\,.\,KK-P,\\
S_{T_5} &\equiv KK-PP + KK-PP\,.\,KKP'ase + KK-PP\,.\,K\\
&\quad + KK-PP\,.\,K-P,\\
S_{T_6} &\equiv K + KK-PP\,.\,K,\\
S_{T_7} &\equiv K-P + K-P\,.\,KP'ase + KK-PP\,.\,K-P,\\
S_{T_8} &\equiv K-PP + K-PP\,.\,KP'ase. \qquad (12)
\end{aligned}$$

Due to the structure of the cascade, the slow-scale propensity functions depend only on the mean of the quasistationary distributions of the virtual fast processes and we exploit this for computational efficiency. In many cases, a good approximation to the mean is afforded by the solution to the corresponding deterministic reaction rate equations (RRE), so we use this approach, which is also used in, for example, the slow-scale SSA.[7] Note that the deterministic tQSSA corresponding to Eq. (12) is used for the RRE approximation and not the usual sQSSA. This is important for accuracy as well as numerical efficiency and stability.

We compute the CME solution at $t_f=10$ in this way, using a matrix of about 12×10^6 in size, in 70 min. This computation is only possible with the help of the tQSSA because, otherwise, the matrix representing the enormous state space of the full model would be too large. The solution is obtained in terms of the total substrates and from this we would normally recover distributions for the other species by using the quasistationary distributions of the virtual fast processes. However, these quasistationary distributions are not available because we only computed approximations to their means, so instead we use these mean values to recover the distribution for the other species.

Figure 6 shows the results for the complex denoted by $KKK.E_1$ that is formed by MAPKKK and the enzyme that activates it, denoted by E_1. The formation of this complex is the critical event that triggers the rest of the signaling cascade. Ciliberto *et al.*[44] suggest that "the role played by enzyme substrate complexes in protein interaction networks could be more important than currently appreciated" based on their observations of the complexes being present in higher than expected concentrations. Our results suggest this finding is also applicable to the MAPK cascade. For example, Fig. 6 shows that the complex species $KKK.E_1$ makes up a relatively large proportion of the available total substrate S_{T_1}, which can be at most 100 for this example.

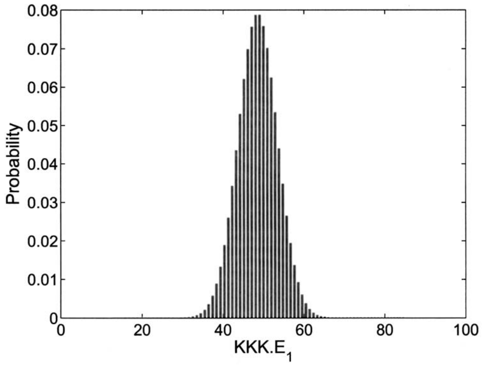

FIG. 7. CME solution at $t_f = 10$ for the MAPK cascade.

For smaller molecular numbers we are able to assess the accuracy of the tQSSA for the MAPK cascade by comparison with the full solution, and the accuracy is quite reasonable, with an error of about 0.01 in the 1-norm. Also for smaller molecular numbers, we can asses the accuracy of the RRE approximation to the means, and these compare extremely favorably to the true means of the quasi-stationary distributions. This gives confidence in the results for the larger example in Fig. 7 and shows that clever and innovative CME-based implementations can be computationally effective on even quite large chemical kinetic problems.

V. DISCUSSION

One of the virtues of PDF approaches, such as the Krylov FSP, is that they provide one way to assess the accuracy of Monte Carlo approaches such as the nSSA. In some cases the tQSSA-based CME solver can be more efficient for the purpose of estimating moments of the distribution. For example, a comparison of the nSSA and the tQSSA CME solver when applied to the double phosphorylation example (e) of Sec. IV B shows a similar trend to that of Fig. 1. On the other hand, systems with very many chemical species, large population numbers, and large propensities provide examples that are better suited to trajectorial methods such as the nSSA. This reflects the inherently high-dimensional nature of the CME, which provides a challenge for all numerical methods. Recently, considerable progress has been made and moderate-sized problems are becoming feasible via various techniques.[12,21,23,34] The tQSSA reduces the dimension of the problem and thus provides yet another way to cope with the curse of dimensionality.

The connection to the theory of aggregation provides a unifying framework for the various versions of the QSSA that have been proposed. In the QSSA, **E** represents the partition of the state space induced by the partition of the reactions into fast and slow, and then **F** is always defined to satisfy Eq. (9). For example, the approximate CME associated with the tQSSA CME solver, the nSSA, and the ssSSA, corresponds to this choice. By allowing other choices for **E** and **F**, as well as using different choices in combination, this connection provides a framework for developing approximations of higher quality or with desirable properties. For example, the choice corresponding to Eq. (9) does not necessarily preserve the stationary solution in the natural way but this can be achieved by varying the choice of **F**.[45]

Lastly, we discuss how to decide when it is appropriate to apply the tQSSA. The slow-scale approximation lemma[17] provides the basis for when the approximation is appropriate. Briefly, the key requirement is that the relaxation time of the virtual fast process must be much smaller than the average time to the next slow reaction.[13] Checking this requirement may involve some analysis but for many systems of interest we already have some intuition that some reactions will be much faster than others. Empirically, the approximation has been demonstrated to perform well for coupled enzymatic networks and signaling cascades.

The tQSSA was described as if the set of fast reactions was fixed. However, examples for which this set must take into account the coupling of reactions and must be changed adaptively and are known.[46] Generalizing the algorithm to allow for the set of fast reactions to change over time, as the system changes state, would allow the approximation to be applied to a wider class of systems. The following is one way to do this.

We examine each "virtual fast process," and check that the slow-scale approximation lemma is satisfied. We compute the ratio of the sum of the propensities of the fast reactions to the sum of the propensities of the slow reactions, and then compare this ratio to a threshold. If the ratio is less than the threshold, we do not use the tQSSA for that region of the state space and instead we simply revert to using the original equations. For those regions of the state space that do pass the test we continue to use the tQSSA.

This approach is analogous to the way that methods such as the ssSSA address the same issue, by regarding the initial partition of reactions as tentative only, checking it dynamically, and then changing it if need be.[13] It corresponds to making a slightly different choice of the aggregation matrix **E**; we simply do not aggregate in those areas of the state space that do not pass the test. This is an example of the benefit of making the connection to the theory of aggregation.

To demonstrate this generalization, we use an example that augments the Michaelis–Menten scheme in Table I, with a fourth reaction,

$$C + P \rightarrow 2P + E.$$

The augmented system may be regarded as an autocatalytic version of the Michaelis-Menten scheme since the formation of the product species, P, now catalyzes the formation of more of itself. The propensity of this extra reaction is $c_4 \times C \times P$, with $c_4 = 0.1$. We set the initial state to be $[S,E,C,P] = [100,10,0,0]$. To begin with, reaction four will be slow, but as the system evolves and the number of molecules of product P grows, the fourth reaction will become very fast. Thus, we have constructed this example specifically so that it has the property that during the dynamical trajectory a reaction may change class. We compute the mean of the number of molecules of the product P at $t_f = 10$, by the generalized algorithm, for various values of the threshold (Table VII), The true solution according to the full CME is also recorded. Using a threshold of zero corresponds

TABLE VII. As the threshold is increased, fewer states are aggregated together so the approximation becomes more accurate. These numerical results, for the augmented Michaelis-Menten system described in Sec. V, paragraph seven, demonstrate this behavior.

Threshold	0	10	100	True
$\mathbb{E}(P)$	977.4	975.2	973.8	973.5

to the original version of our algorithm for which the set of fast reactions is fixed and it can be seen that this approach is not very accurate. This is just what we expect for an example such as this that was designed specifically to illustrate this point. However, as the threshold is increased, the accuracy improves considerably because the tQSSA is only applied to those regions of the state space for which the approximation is suitable. More sophisticated strategies will be developed in future work but even this simple example illustrates that our algorithm can be generalized to accommodate adaptive partitioning of the reactions.

VI. CONCLUSIONS

The total QSSA has been generalized to the stochastic setting by making some important connections to the literature on aggregation, resulting in a CME solver that is more computationally efficient. The new methods have been successfully demonstrated on Michaelis–Menten enzyme kinetics, double phosphorylation, the Goldbeter–Koshland switch, and the MAPK cascade. Overall, the application of the tQSSA CME solver was extremely successful since it dramatically reduces the size of the problem and speeds up the computation very considerably, while maintaining acceptable accuracy.

APPENDIX: THE tQSSA FOR MICHAELIS–MENTEN ENZYME KINETICS

We give a formula for the quasiequilibrium distributions of the virtual fast processes. For each rectangle corresponding to $S_T=0,1,\ldots,S_I$, let $\hat{P}(S|S_T)$ denote the probability of the state with S substrates, according to the quasiequilibrium distribution.

Let $S_{\min}\equiv\max(0,S_T-E_I)$ and $\hat{P}(S_{\min}|S_T)\equiv 1$. Then $\hat{P}(\cdot|S_T)$ (not yet normalized) is defined recursively, for $S=S_{\min},\ldots,S_T-1$, by $\hat{P}(S+1|S_T)=K(S_T-S)/(E_I-[S_T-(S+1)])(S+1)\hat{P}(S|S_T)$, where $K\equiv c_2/c_1$. The slow-scale propensities for the reduced model are $\alpha(S_T\rightarrow S_T-1)=c_3\mathbb{E}[C|S_T]=c_3A_{S_T}\sum_{S=S_{\min}}^{S_T-1}(S_T-S)\hat{P}(S|S_T)$, for $S_T=1,\ldots,S_I$. Here, $A_{S_T}\equiv(\sum_{S=S_{\min}}^{S_T}\hat{P}(S|S_T))^{-1}$ is a normalization constant.

Enumerating the states in increasing order of the number of products, the matrix $\mathbf{B}$ is bidiagonal, of size S_I+1, with $b_{ii}\equiv-b_{i+1,i}$ and $b_{i+1,i}\equiv\alpha(S_I-i+1\rightarrow S_I-i)$. The last column is zero, corresponding to the absorbing state. Thus, the tQSSA reduces this example to a one-dimensional, pure death process. The sQSSA makes the same approximation without introducing the change of variables, which in the stochastic setting corresponds to aggregating states with the same number of free substrates. The states within a block of such a partition are connected by the relatively slow reactions, so this approximation corresponds to the counterintuitive assumption that the slow reactions almost reach equilibrium before the fast reactions take effect. Thus, by itself, this stochastic version of the sQSSA is not a sensible approximation. However, the two approximations may be used in combination to achieve more accurate results than either of them used alone, in analogy with operator splitting.[47]

We now compare the tQSSA with Rao and Arkin's interpretation of the QSSA as applied to the CME associated with Michaelis–Menten enzyme kinetics. Although the total substrate variable $S_T=S+C$ is explicitly introduced in Ref. 20, p. 5002, there is no connection to the tQSSA. In fact, Eq. (17) in Ref. 20 is obtained by performing an asymptotic expansion of the probability P in terms of the perturbing term $\epsilon := e_0/s_0$. This mechanism, with the same parameter, was suggested by Heineken *et al.*,[48] in a deterministic framework, in order to show that the sQSSA can be considered as the zero-order approximation of the system. The asymptotic expansion proposed by Rao and Arkin is valid only if $e_0/s_0 \ll 1$. Thus, their approximation cannot be classified as an example of the tQSSA. As a confirmation of this fact, in Fig. 1 of Ref. 20, it is reported that the approximation performs well in the case where substrates are in excess of enzymes but performs poorly when the situation is reversed. This is in contrast to the deterministic setting where the tQSSA was introduced precisely because it performed better for the case where enzymes were in excess. As a partial justification of the poor performance of their approximation in the stochastic setting, when enzymes are in excess, Fig. 3 of Ref. 20 shows that, in the deterministic setting, the QSSA also performs poorly. This highlights the need to distinguish between the various forms of the QSSA and make connections to the deterministic literature because, for example, the tQSSA would perform well in both regimes.

[1] W. J. Blake, M. Kærin, C. R. Cantor, and J. J. Collins, Nature (London) **422**, 633 (2003).

[2] N. Fedoroff and W. Fontana, Science **297**, 1129 (2002).

[3] N. G. van Kampen, *Stochastic Processes in Physics and Chemistry* (Elsevier Science, New York, 2001).

[4] D. T. Gillespie, *Markov Processes: An Introduction for Physical Scientists* (Academic, New York, 1992).

[5] A. Arkin, J. Ross, and H. McAdams, Genetics **149**, 1633 (1998).

[6] D. T. Gillespie, J. Phys. Chem. **81**, 2340 (1977).

[7] D. T. Gillespie, J. Chem. Phys. **115**, 1716 (2001).

[8] K. Burrage, S. Mac, and T. Tian, Lect. Notes Control Inf. Sci. **34**, 359 (2006).

[9] M. Rathinam, L. R. Petzold, Y. Cao, and D. T. Gillespie, Multiscale Model. Simul. **4**, 867 (2005).

[10] T. Tian and K. Burrage, J. Chem. Phys. **121**, 10356 (2004).

[11] K. Burrage, T. Tian, and P. Burrage, Prog. Biophys. Mol. Biol. **85**, 217 (2004).

[12] P. Lötstedt and L. Ferm, Multiscale Model. Simul. **5**, 593 (2006).

[13] Y. Cao, D. T. Gillespie, and L. R. Petzold, J. Chem. Phys. **122**, 014116 (2005).

[14] Weinan E, D. Liu, and E. Vanden-Eijnden, J. Chem. Phys. **123**, 194107 (2005).

[15] Weinan E, D. Liu, and E. Vanden-Eijnden, J. Comput. Phys. **221**, 158 (2007).

[16] Weinan E, D. Liu, and E. Vanden-Eijnden, J. Chem. Phys. **126**, 137102 (2007).

[17] D. T. Gillespie, L. R. Petzold, and Y. Cao, J. Chem. Phys. **126**, 137101 (2007).

[18] J. Goutsias, J. Chem. Phys. **122**, 184102 (2005).

[19] E. L. Haseltine and J. B. Rawlings, J. Chem. Phys. **117**, 6959 (2002).
[20] C. V. Rao and A. P. Arkin, J. Chem. Phys. **118**, 4999 (2003).
[21] K. Burrage, M. Hegland, S. MacNamara, and R. B. Sidje, *150th Proceedings of the Markov Anniversary Meeting, Charleston, SC, USA*, edited by A. A. Langville and W. J. Stewart (Boson Books, Raleigh, 2006), pp. 21–38.
[22] S. MacNamara, K. Burrage, and R. B. Sidje, ANZIAM J. **48**, C413 (2007).
[23] S. MacNamara, K. Burrage, and R. B. Sidje, Multiscale Model. Simul. **6**, 146 (2008).
[24] M. B. Elowitz, M. G. Surette, P. E. Wolf, J. B. Stock, and S. Leibler, J. Bacteriol. **181**, 197 (1999).
[25] D. Nicolau, Jr., K. Burrage, R. G. Parton, and J. Hancock, Mol. Cell. Biol. **26**, 313 (2006).
[26] M. Rathinam, L. R. Petzold, Y. Cao, and D. T. Gillespie, J. Chem. Phys. **119**, 12784 (2003).
[27] B. Munsky and M. Khammash, J. Chem. Phys. **124**, 044104 (2006).
[28] R. B. Sidje, EXPOKIT, a software package for computing matrix exponentials, www.expokit.org; ACM Trans. Math. Softw. **24**, 130 (1998).
[29] R. B. Sidje and W. J. Stewart, Comput. Stat. Data Anal. **29**, 345 (1999).
[30] R. I. McLachlan and G. R. W. Quispel, Acta Numerica **11**, 341 (2002).
[31] S. Peleš, B. Munsky, and M. Khammash, J. Chem. Phys. **125**, 204104 (2006).
[32] G. Strang, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **5**, 506 (1968).
[33] Y. Saad, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **29**, 209 (1992).
[34] M. Hegland, C. Burden, L. Santoso, S. MacNamara, and H. Booth, J. Comput. Appl. Math. **205**, 708 (2007).
[35] W. J. Stewart, *Introduction to the Numerical Solution of Markov Chains* (Princeton University Press, Princeton, 1994).
[36] R. H. Chan, Numer. Math. **51**, 143 (1987).
[37] J. A. M. Borghans, R. J. De Boer, and L. A. Segel, Bull. Math. Biol. **58**, 43 (1996).
[38] M. G. Pedersen, A. M. Bersani, and E. Bersani, Bull. Math. Biol. **69**, 443 (2007).
[39] M. G. Pedersen, A. M. Bersani, and E. Bersani, J. Math. Chem. **4**, 1318 (2008).
[40] M. G. Pedersen, A. M. Bersani, E. Bersani, and G. Cortese, *Proceedings of the 5th MATHMOD Conference*, ARGESIM Report No. 30 (Vienna University of Technology Press, Vienna, 2006).
[41] A. R. Tzafrifri, Bull. Math. Biol. **65**, 1111 (2003).
[42] C.-Y. F. Huang and J. E. Ferrell, Jr., Proc. Natl. Acad. Sci. U.S.A. **93**, 10078 (1996).
[43] A. Goldbeter and D. E. Koshland, Proc. Natl. Acad. Sci. U.S.A. **78**, 6840 (1981).
[44] A. Ciliberto, F. Capuani, and J. J. Tyson, PLOS Comput. Biol. **3**, 0463 (2007).
[45] C. J. Burke and M. Rosenblatt, Ann. Math. Stat. **29**, 1112 (1958).
[46] M. F. Pettigrew and H. Resat, J. Chem. Phys. **126**, 084101 (2007).
[47] S. MacNamara, K. Burrage, and R. B. Sidje, Int. J. Comput. Sci. **3**, 402 (2008).
[48] F. A. Heineken, H. M. Tsuchiya, and R. Aris, Math. Biosci. **1**, 95 (1967).