HYBRID METHOD FOR THE CHEMICAL MASTER EQUATION

ANDREAS HELLANDER[1], PER LÖTSTEDT[1]

[1]*Division of Scientific Computing, Department of Information Technology*
Uppsala University, SE-75105 Uppsala, Sweden
emails: `andreas.hellander@it.uu.se, perl@it.uu.se`

Abstract

The chemical master equation is solved by a hybrid method coupling a macroscopic, deterministic description with a mesoscopic, stochastic model. The molecular species are divided into one subset where the expected values of the number of molecules are computed and one subset with species with a stochastic variation in the number of molecules. The macroscopic equations resemble the reaction rate equations and the probability distribution for the stochastic variables satisfy a master equation. The probability distribution is obtained by the Stochastic Simulation Algorithm due to Gillespie. The equations are coupled via a summation over the mesoscale variables. This summation is approximated by Monte Carlo and Quasi Monte Carlo methods. The error in the approximations is analyzed. The hybrid method is applied to three chemical systems from molecular cell biology.

Keywords: master equation, reaction rate equations, stochastic simulation algorithm, Monte Carlo method, Quasi Monte Carlo method, stochastic chemical kinetics

AMS subject classification: 65M20, 65M60

1   Introduction

In a well stirred chemical system, the chemical reactions are often modeled by the reaction rate equations. These equations form a system of non-linear, coupled ordinary differential equations (ODEs). Such a macroscopic model provides a good description of the time evolution for the concentrations of the chemcial species of the system in many cases e.g. when the number of molecules of each kind is large and in the absence of critical phenomena. On the other hand, the species

with low copy number are not well described by a deterministic and macroscopic model since they are subject to random fluctuations which cannot be neglected and in many cases have a great impact on the behavior of the system. In a biological cell, the underlying assumptions for the reaction rate equations are often violated [1, 4, 24, 27, 30, 31, 33, 34]. At least some species are usually present in low copy numbers. For example, mRNA usually exists in one or a few copies, while transcription factors may be present in the range from ten to hundreds of molecules. Yet other components could be active in large numbers and approach macroscopic values. Thus, a realistic model must take the inherent randomness into account and therefore need to be of stochastic nature. A disadvantage with stochastic models is the increase in computational complexity compared to the reaction rate equations. Another source of computational difficulties is the different scales both in time and in reaction rates.

One way to model coupled chemical reactions stochastically at a mesoscopic level is to the use the Stochastic Simulation Algorithm (SSA) proposed by Gillespie [10]. This Monte Carlo algorithm yields a correct realization of the process, but the computing time required to approximate the probability distribution of the species in the system is often dictated by the reactions involving the molecules with the largest copy numbers or the fastest reaction rates. They may well be the components where the stochastic description is the least important. The convergence rate is also slow for this method and it can be computationally cumbersome to obtain detailed information of the probability distributions when the number of different reacting molecules is large.

The underlying stochastic process is often assumed to be memory lacking or *Markovian*. Then the time evolution of the probability distribution is described by a difference-differential equation, the chemical master equation [9, 18]. One molecular species corresponds to one spatial dimension in the equation. Analytical solutions of this equation are known only for very simple chemical systems and numerical solution is necessary for realistic systems. Direct numerical solution of the master equation suffers from the curse of dimensionality as the computational work and storage requirements grow exponentially with the number of dimensions or reacting species. Consequently, this often limits the size of the models to four or maybe five dimensions.

Different ways to avoid the exponential growth have been proposed, either by approximations of the master equation [32] or dimension reduction by introducing assumptions about the behavior of different components [34]. In the first case, the master equation is approximated by the Fokker-Planck equation, a partial differential equation derived from a truncated Taylor expansion of the master equation [9, 18]. The discretized Fokker-Planck equation can be solved with fewer variables compared to the master equation, but this approach is still limited by an unfavorable rise in computational time with increasing number of species. The second approach relies on some previous knowledge of the system in order to reduce the dimension of the problem. While this can result in a considerable

reduction of the complexity, a profound knowledge of the biological system is required to introduce some simplifying assumptions. For moderate numbers of reacting species, sparse grid methods are an alternative [15].

With SSA [10] the work grows linearly with the number of species but for systems with different time scales the method is slow owing to the explicit time stepping. By allowing more reactions to take place in a time step or assuming approximate relations between the species, longer time steps are possible [3, 11, 28, 29]. There are efficient numerical methods to solve ODEs and the fast components are modeled by deterministic equations and other components are treated with a stochastic model in [14, 19].

A separation of the chemical compounds into a subset of variables that can be treated as normally distributed with a small variance and a subset of variables that need a stochastic treatment is suggested in [22]. Equations are derived for the expected values of the first subset. They can be solved given the probability density function (PDF) of the stochastic variables. This PDF satisfies a Fokker-Planck equation and is solved by a finite volume scheme in [22] but the dimension of the stochastic problem is then restricted to, say, five. Here, we apply SSA to the stochastic part and compute the coupling to the deterministic part using Monte Carlo and Quasi Monte Carlo summation [2, 12]. The equations for the expected values are integrated in time by an unconditionally stable, implict method. Hence, if the species in the fast reactions are treated macroscopically, then the small time steps in SSA are avoided as in [14, 19]. Our method can be regarded either as a means to introduce stochasticity in some components in the reaction rate equations or as a way of improving the efficiency of SSA by reducing the number of species in its system state vector.

In the next section, the nonlinear system of differential equations for the expected values and the master equation for the PDF of the stochastic variables are derived. The solution algorithm for the coupled system is described in Section 3. The errors in the solution are discussed and the computational work is estimated in Section 4. Three systems in molecular biology are simulated in Section 5 using our algorithm. In one example, the behavior of the macroscopic model is sensitive to a parameter but addition of stochastic variables removes this sensitivity. In another example, the full SSA simulation is compared to a mixed macroscopic-mesoscopic model. The difference in the probability distribution of critical components is small while the savings in computational work are significant. Finally, conclusions are drawn in the last section.

The notation in the paper is as follows. The i:th element of a vector $\mathbf{v}$ is denoted by v_i. If $v_i \geq 0$ for all i, then we write $\mathbf{v} \geq 0$. The ℓ_p-norm of $\mathbf{v}$ of length N is $\|\mathbf{v}\|_p = (\sum_{i=1}^{N} |v_i|^p)^{1/p}$. The set of integer numbers is written $\mathbb{Z}$ and $\mathbb{Z}_+$ denotes the non-negative integer numbers. In the same manner, $\mathbb{R}$ denotes the real numbers and $\mathbb{R}_+$ is the non-negative real numbers.

2 The system of equations

Assume that we have a chemical system with N active molecular species X_i, $i = 1, \ldots, N$, and that x_i denotes the number of molecules of substrate X_i. The system has a state vector $\mathbf{x} \in \mathbb{Z}_+^N$, and a reaction r in the system is a transition from a state $\mathbf{x}_r$ to $\mathbf{x}$ so that $\mathbf{x}_r = \mathbf{x} + \mathbf{n}_r$ with $\mathbf{n}_r \in \mathbb{Z}^N$. Only a few components of $\mathbf{n}_r$ are nonzero. The probability of the reaction to occur per unit time is the non-negative propensity $w_r(\mathbf{x}_r, t)$. The change in the state vector by a reaction r can now be written

$$\mathbf{x}_r \xrightarrow{w_r(\mathbf{x}_r,t)} \mathbf{x}, \ \mathbf{n}_r = \mathbf{x}_r - \mathbf{x}. \tag{1}$$

The PDF $p(\mathbf{x}, t)$ for the system to be in the state $\mathbf{x}$ at time t satisfies the chemical master equation [9, 18]. With a splitting of $\mathbf{n}_r$ into two parts so that

$$\mathbf{n}_r = \mathbf{n}_r^+ + \mathbf{n}_r^-, n_{ri}^+ = \max(n_{ri}, 0), n_{ri}^- = \min(n_{ri}, 0),$$

the master equation for R reactions is

$$\frac{\partial p(\mathbf{x}, t)}{\partial t} = \sum_{\substack{r = 1 \\ \mathbf{x} + \mathbf{n}_r^- \geq 0}}^{R} w_r(\mathbf{x} + \mathbf{n}_r, t)p(\mathbf{x} + \mathbf{n}_r, t) - \sum_{\substack{r = 1 \\ \mathbf{x} - \mathbf{n}_r^+ \geq 0}}^{R} w_r(\mathbf{x}, t)p(\mathbf{x}, t). \tag{2}$$

It follows from [8] that the total probability $\sum_{\mathbf{x} \in \mathbb{Z}_+^N} p(\mathbf{x}, t)$ is constant in time.

In order to reduce the computational complexity of solving (2), $\mathbf{x}$ is split in [22] into two parts $\mathbf{x}^T \to (\mathbf{x}^T, \mathbf{y}^T)$ with $\mathbf{x} \in \mathbb{Z}_+^m$, $\mathbf{y} \in \mathbb{R}^n$, and $N = m + n$. In the same manner, the transition vector $\mathbf{n}_r$ is split $\mathbf{n}_r^T \to (\mathbf{m}_r^T, \mathbf{n}_r^T)$ for reaction r. The dimensions of $\mathbf{m}_r$ and $\mathbf{n}_r$ are $\mathbf{m}_r \in \mathbb{Z}_+^m$, $\mathbf{n}_r \in \mathbb{Z}_+^n$. The corresponding stochastic variables are $X_i, i = 1, \ldots, m$, and $Y_i, i = 1, \ldots, n$. The assumption is that the stochastic variables Y_i are mutually independent, independent of X_i, and normally distributed with a small variance. Then the PDF of the full system is written

$$p(\mathbf{x}, \mathbf{y}, t) = \gamma_n p_0(\mathbf{x}, t) \exp\left(-\sum_{j=1}^{n} \frac{(y_j - \phi_j(t))^2}{2\sigma_j^2}\right), \tag{3}$$

with the normalizing constant $\gamma_n = (2\pi)^{-n/2} \prod_{j=1}^{n} \sigma_j^{-1}$. Equations will be derived for p_0 and $\boldsymbol{\phi} \in \mathbb{R}^n$ such that p approximately fulfills (2).

The marginal PDF of p in (3) satisfies

$$p_0(\mathbf{x}, t) = \int p(\mathbf{x}, \mathbf{y}, t) \, d\mathbf{y},$$

where the domain of the integral is $\mathbb{R}^n$. The scaling of p_0 is such that the total probability satisfies

$$\sum_{\mathbf{x} \in \mathbb{Z}_+^m} \int p(\mathbf{x}, \mathbf{y}, t) = \sum_{\mathbf{x} \in \mathbb{Z}_+^m} p_0(\mathbf{x}, t) = 1. \tag{4}$$

The expected value of Y_k is

$$E[Y_k] = \sum_{\mathbf{x} \in \mathbb{Z}_+^m} \int y_k p(\mathbf{x}, \mathbf{y}, t) \, d\mathbf{y} = \phi_k(t).$$

A differential-difference equation for p_0 is derived in [22] assuming that $\sigma_j \leq \sigma$ and that σ is small. Ignoring terms proportional to σ^2 for a vanishing σ, the equation is for R reactions

$$\frac{\partial p_0(\mathbf{x}, t)}{\partial t} = \sum_{\substack{r=1 \\ \mathbf{x} + \mathbf{m}_r^- \geq 0}}^{R} w_r(\mathbf{x} + \mathbf{m}_r, \boldsymbol{\phi}(t), t) p_0(\mathbf{x} + \mathbf{m}_r, t) \tag{5}$$
$$- \sum_{\substack{r=1 \\ \mathbf{x} - \mathbf{m}_r^+ \geq 0}}^{R} w_r(\mathbf{x}, \boldsymbol{\phi}(t), t) p_0(\mathbf{x}, t).$$

This is a master equation for p_0 with propensities depending on the expected values $\boldsymbol{\phi}$. It follows from (5) that the total marginal probability $\sum_{\mathbf{x} \in \mathbb{Z}_+^m} p_0(\mathbf{x}, t)$ is constant (cf. (2)) as presupposed in (4).

The differential equation for ϕ_j, $j = 1, \ldots, n$, is for a single reaction, $R = 1$,

$$\begin{aligned}
\frac{d\phi_j}{dt} &= \sum_{\mathbf{x} \in \mathbb{Z}_+^m} \int y_j \frac{\partial p}{\partial t} \, d\mathbf{y} \\
&= \sum_{\mathbf{x} + \mathbf{m}_r^- \geq 0} w_r(\mathbf{x} + \mathbf{m}_r, \boldsymbol{\phi}(t), t) p_0(\mathbf{x} + \mathbf{m}_r, t)(\phi_j - n_{rj}) \\
&\quad - \sum_{\mathbf{x} - \mathbf{m}_r^+ \geq 0} w_r(\mathbf{x}, \boldsymbol{\phi}(t), t) p_0(\mathbf{x}, t) \phi_j \\
&= \phi_j \sum_{\mathbf{x} \in \mathbb{Z}_+^m} \frac{\partial p_0(\mathbf{x}, t)}{\partial t} - n_{rj} \sum_{\mathbf{x} + \mathbf{m}_r^- \geq 0} w_r(\mathbf{x} + \mathbf{m}_r, \boldsymbol{\phi}(t), t) p_0(\mathbf{x} + \mathbf{m}_r, t) \\
&= -n_{rj} \sum_{\mathbf{x} + \mathbf{m}_r^- \geq 0} w_r(\mathbf{x} + \mathbf{m}_r, \boldsymbol{\phi}(t), t) p_0(\mathbf{x} + \mathbf{m}_r, t),
\end{aligned} \tag{6}$$

by (2), (5), the conservation of the total marginal probability, and after ignoring small terms of $\mathcal{O}(\sigma^2)$, see [22]. The equation is simplified by removing the shift $\mathbf{m}_r$ so that for R reactions

$$\frac{d\phi_j}{dt} = -\sum_{r=1}^{R} n_{rj} \sum_{\mathbf{x} \in \mathbb{Z}_+^m} w_r(\mathbf{x}, \boldsymbol{\phi}(t), t) \, p_0(\mathbf{x}, t), \quad j = 1, \ldots, n. \tag{7}$$

This system of differential-summation equations is equal to the reaction rate equations for the chemical system when the distribution of all species is assumed to be normal with small variances.

5

Suppose that the mesoscopic $\mathbf{X}$ participates in only the first ρ reactions. Then $\mathbf{m}_r = 0$, $r = \rho + 1, \ldots, R$, and the right hand side in (5) is reduced to

$$
\begin{aligned}
&\sum_{\substack{r=1 \\ \mathbf{x}+\mathbf{m}_r^- \,\geq\, 0}}^{\rho} w_r(\mathbf{x} + \mathbf{m}_r, \boldsymbol{\phi}(t), t) p_0(\mathbf{x} + \mathbf{m}_r, t) \\
&- \sum_{\substack{r=1 \\ \mathbf{x}-\mathbf{m}_r^+ \,\geq\, 0}}^{\rho} w_r(\mathbf{x}, \boldsymbol{\phi}(t), t) p_0(\mathbf{x}, t).
\end{aligned}
\tag{8}
$$

If $\rho = 0$, then $\partial p_0/\partial t = 0$ and $p_0(\mathbf{x}, t) = p_0(\mathbf{x}, 0)$. The summation over $\mathbf{x}$ in (7) should then be performed in a preparatory stage before the simulation starts.

Suppose that w_r depends on $\mathbf{x}$ only in the κ first reactions. Then the right hand side in (7) is

$$
\begin{aligned}
&-\sum_{r=1}^{\kappa} n_{rj} \sum_{\mathbf{x}\in\mathbb{Z}_+^m} w_r(\mathbf{x}, \boldsymbol{\phi}(t), t)\, p_0(\mathbf{x}, t) - \sum_{r=\kappa+1}^{R} n_{rj} \sum_{\mathbf{x}\in\mathbb{Z}_+^m} w_r(\boldsymbol{\phi}(t), t)\, p_0(\mathbf{x}, t) = \\
&-\sum_{r=1}^{\kappa} n_{rj} \sum_{\mathbf{x}\in\mathbb{Z}_+^m} w_r(\mathbf{x}, \boldsymbol{\phi}(t), t)\, p_0(\mathbf{x}, t) - \sum_{r=\kappa+1}^{R} n_{rj} w_r(\boldsymbol{\phi}(t), t).
\end{aligned}
\tag{9}
$$

If $\kappa = 0$, then (7) are the reaction rate equations.

Two equations have been derived for p_0 in (5) and for ϕ_j in (7). They are coupled via the expected values of the propensity of each reaction in (7) and the expected values of the species in the propensities in (5). The dimension of the stochastic problem has been reduced from N in (2) to m in (5). The cost for this reduction is the increased number of dependent variables from 1 in (2) to $n + 1$ in (5) and (7) but the work grows at most as a polynomial in n solving (5) and (7) compared to an exponential growth in n when is (2) solved.

3 Solution algorithm

The two systems of equations (5) and (7) are solved numerically by a hybrid method where p_0 in (5) is determined by SSA and $\boldsymbol{\phi}$ in (7) is computed by a deterministic time stepping method.

3.1 Solution of the master equation

The SSA is applied to the chemical system defined by the master equation for p_0 in (5) with the right hand side (8). Trajectories of the system are simulated by

updating the state vectors after each reaction in [10]. Then the probability for the system to be in a state $\mathbf{x}$ at time t^ν is approximated by

$$p_0(\mathbf{x}, t^\nu) \approx \frac{1}{M} \sum_{j=1}^{M} \Psi_j, \qquad \Psi_j = \begin{cases} 1, & \mathbf{x}_j = \mathbf{x}, \\ 0, & \text{otherwise,} \end{cases} \qquad (10)$$

where M is the number of trajectories. The error in p_0 due to the finite M is analyzed in Sect. 4.1

The time evolution of the state is simulated by SSA in the following way. Let w^ν be the sum of the propensities at t^ν so that

$$w^\nu = \sum_{r=1}^{\rho} w_r(\mathbf{x}^\nu, \boldsymbol{\phi}(t^\nu), t^\nu), \qquad (11)$$

cf. (8). In the direct method in [10], the next reaction after t^ν occurs at $t^\nu + \Delta\tau^\nu$, where the time increment $\Delta\tau^\nu$ is exponentially distributed with mean $1/w^\nu$. If the partitioning of the species into $\mathbf{x}$ and $\mathbf{y}$ is chosen such that $\mathbf{m}_r = 0$ for the largest propensities w_r, i.e. w_r is small for $r \leq \rho$, then the expected value of $\Delta\tau^\nu$ is longer than the time step without partitioning making SSA more efficient. The reaction number μ with $1 \leq \mu \leq R$ is chosen with probability $w_\mu(\mathbf{x}^\nu, \boldsymbol{\phi}(t^\nu), t^\nu)/w^\nu$. With the stoichiometric matrix S defined by

$$S = (\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_\rho)$$

and the unit vector $\mathbf{e}_j \in \mathbb{R}^\rho$ with 1 in the j:th position, the algorithm can be written as a stochastic process for each trajectory

$$\mathbf{X}^{\nu+1} = \mathbf{X}^\nu - S\mathbf{e}_\mu, \quad t^{\nu+1} = t^\nu + \Delta\tau^\nu. \qquad (12)$$

The M state vectors $\mathbf{X}_j^\nu$ are stored in a trajectory matrix $\mathbf{T}^\nu$ such that $\mathbf{T}_{j\cdot}^\nu = (\mathbf{X}_j^\nu)^T$, $j = 1, \ldots, M$. The evaluation of p_0 in (10) for a given $\mathbf{x}$ is simplified if the rows of the matrix are sorted such that $\mathbf{T}_{j1}^\nu$ increases for increasing j. Then each interval in j with a constant $\mathbf{T}_{j,l}^\nu$ is sorted in ascending values of $\mathbf{T}_{j,l+1}^\nu$ for $l = 1, \ldots, M-1$. Identical rows are removed and the frequency of the row is saved. Then the row in $\mathbf{T}^\nu$ corresponding to a particular $\mathbf{x}_k$ is easily found by binary search.

3.2 Solution of the differential-summation equation

The time derivative of the expected value in (7) is approximated by an implicit backward differentiation formula of order 2 (BDF2) with variable time steps [13]. The increment between t^k and t^{k+1} is denoted by Δt^k. Then the new value of

ϕ^{k+1} at t^{k+1} is computed by

$$
\begin{aligned}
&\alpha_0^k \phi^{k+1} = \Delta t^k \mathbf{F}(\phi^{k+1}, t^{k+1}) - \alpha_1^k \phi^k - \alpha_2^k \phi^{k-1}, \\
&\boldsymbol{\omega}(\mathbf{x}, \phi, t) = -\sum_{r=1}^{R} \mathbf{n}_r w_r(\mathbf{x}, \phi, t), \\
&F_j(\phi, t^{k+1}) = -\sum_{\mathbf{x}^{k+1} \in \mathbb{Z}_+^m} \omega_j(\mathbf{x}^{k+1}, \phi, t^{k+1}) \, p_0(\mathbf{x}^{k+1}, t^{k+1}), \ j = 1, \ldots, n, \\
&\alpha_0^k = (1 + 2\theta^k)/(1 + \theta^k), \ \alpha_1^n = -(1 + \theta^k), \ \alpha_2^k = (\theta^k)^2/(1 + \theta^k), \\
&\theta^k = \Delta t^k / \Delta t^{k-1},
\end{aligned}
\tag{13}
$$

see [13, 23]. A predicted value $\hat{\phi}^{k+1}$ is computed with an explicit scheme using

$$
\begin{aligned}
&\hat{\alpha}_0^k \hat{\phi}^{k+1} = \Delta t^k \mathbf{F}(\phi^n, t^n) - \hat{\alpha}_1^k \phi^n - \hat{\alpha}_2^k \phi^{k-1}, \\
&\hat{\alpha}_0^{\ k} = 1/(1 + \theta^k), \ \hat{\alpha}_1^{\ n} = \theta^k - 1, \ \hat{\alpha}_2^{\ k} = -(\theta^k)^2/(1 + \theta^k),
\end{aligned}
\tag{14}
$$

see [23]. The local temporal discretization error τ^{k+1} in the difference approximation is proportional to $(\Delta t^k)^2$. It is estimated by subtracting $\hat{\phi}^{k+1}$ from ϕ^{k+1} and dividing by Δt^k. The next time step Δt^{k+1} is chosen after comparing τ^{k+1} with an error tolerance so that τ^{n+2} is expected to satisfy the tolerance in the next step, see Sect. 4.4.

The system of nonlinear equations in (13) satisfied by ϕ^{k+1} is solved by Newton iterations. The initial guess of ϕ^{k+1} in the iterations is $\hat{\phi}^{k+1}$. The elements of the Jacobian $\mathbf{J}$ in Newton's method is then computed as

$$
\begin{aligned}
&J_{ij} = \left(\sum_{\mathbf{x} \in \mathbb{Z}_+^m} \omega_i(\mathbf{x}, \phi + \mathbf{e}_j \Delta \phi_j, t) p_0(\mathbf{x}, t) - \sum_{\mathbf{x} \in \mathbb{Z}_+^m} \omega_i(\mathbf{x}, \phi, t) p_0(\mathbf{x}, t) \right) / \Delta \phi_j, \\
&i, j = 1, \ldots, n,
\end{aligned}
\tag{15}
$$

i.e. using first order forward differences. The second summation over $\mathbf{x}$ needs to be evaluated only once with this approximation of the derivatives. The number of participating species in every reaction is small implying that $\mathbf{J}$ is rather sparse. This fact or if κ is small in (9) makes the summation over $\mathbf{x}$ necessary only for a small fraction of $\mathbf{J}$. To take advantage of this, the system specification includes a dependency matrix $\mathbf{D} \in \mathbb{Z}^{n \times n}$, in which the elements D_{ij} take the integer value 1 when ω_i in (13) depends on the chemical species j and 0 otherwise. The summation over the stochastic variables is necessary only for J_{ij} with $D_{ij} = 1$. The Jacobian is stored in factorized form and is recomputed only if the convergence of the iterations is too slow.

The stability of the integration in (13) is insensitive to the stiffness of $\boldsymbol{\omega}$ with respect to ϕ. If possible, species involved in reactions with propensities potentially contributing to the stiffness of $\mathbf{J}$ should be treated deterministically in ϕ.

The time steps in (12) and (13) are not necessarily syncronized so that $t^{\nu+1} = t^{k+1}$. The PDF p_0 in (13) is calculated from (10) where all the trajectories have been advanced by (12) so that $t^{\nu+1} \leq t^{k+1}$ but in the next step $t^{\nu+2} > t^{k+1}$. The value of ϕ^ν in (11) at t^ν, $t^k < t^\nu \leq t^{k+1}$ is taken to be constant during the SSA steps and equals the value computed at t^k.

3.3 Approximation of the summation

The sums over $\mathbf{x}$ in (13) and (15) can be expensive to compute with exponential growth of the work in the dimension of $\mathbf{x}$ even if the summation is restricted to a finite but still multidimensional domain in $\mathbb{Z}_+^m$ where $p_0 \neq 0$. An alternative to summation over $\mathbb{Z}_+^m$ is to approximate the sums by a Monte Carlo (MC) or a Quasi Monte Carlo (QMC) approach.

Let $\Omega = \{\mathbf{x} \,|\, 0 \leq x_i \leq x_{\max}\} \subset \mathbb{Z}_+^m$ for some $x_{\max} > 0$ and assume that $p_0 = 0$ outside Ω. Then in (13)

$$
\sum_{\mathbf{x} \in \Omega} \boldsymbol{\omega}(\mathbf{x}, \boldsymbol{\phi}, t)\, p_0(\mathbf{x}, t) = E[\boldsymbol{\omega}(\mathbf{X}_0, \boldsymbol{\phi}, t)] \approx \frac{\zeta}{K} \sum_{k=1}^{K} \boldsymbol{\omega}(\mathbf{x}_k, \boldsymbol{\phi}, t)\, p_0(\mathbf{x}_k, t), \qquad (16)
$$

where $\mathbf{X}_0$ has the PDF p_0. Let $x_{j,min}$ and $x_{j,max} \leq x_{\max}$ be such that $p_0 = 0$ if $x_j < x_{j,min}$ or $x_j > x_{j,max}$ for every j. Then the factor in (16) is

$$
\zeta = \prod_{j=1}^{\rho} ((x_{j,max} + 1) - x_{j,min}).
$$

The K quadrature points $\mathbf{x}_k$ are chosen by an MC or QMC method.

In a standard MC method, the sum in (16) is evaluated by generating pseudorandom vectors $\mathbf{x}_k$ with x_{ki} having uniform distribution in $[x_{i,min}, x_{i,min} + 1, \ldots, x_{i,max}]$. The method converges slowly with the rate $K^{-1/2}$ but independently of m and the regularity of $\boldsymbol{\omega} p_0$ [2, 12].

An alternative is to use QMC methods where the sequence of quasirandom $\mathbf{x}_k$ is generated deterministically. For x_{ki}, a quasirandom number ξ in $[0, 1)$ is first generated according to some rule. Then it is scaled so that x_{ki} is the integer part of $x_{j,min} + ((x_{j,max} + 1) - x_{j,min})\xi$. The convergence rate for smooth functions is now of $\mathcal{O}(K^{-1}(\log K)^m)$, which is an improvement compared to MC at least for a moderate m [2]. The summation errors in MC and QMC are estimated in Sects. 4.2 and 4.3.

The sum (16) can be computed by an acceptance-rejection method [2, 12] where a uniformly distributed trial point $\mathbf{x}_k$ is accepted in the sum with probability proportional to p_0. Then the summands can be written $v_k \omega_j(\mathbf{x}_k, \boldsymbol{\phi}, t)$ where the weights v_k are one or zero depending on if the point is accepted or rejected. The convergence rate of QMC depends on the smoothness of the summand and it may deteriorate because of the lack of regularity in $v_k \omega_j(\mathbf{x}_k, \boldsymbol{\phi}, t)$. By introducing

a linear approximation of v_k close to the switch from 1 to 0, better convergence is achieved in [2, 25]. The convergence of the MC and QMC methods are compared using Faure sequences [6] for QMC in a numerical example in Sect. 5.

4   Analysis of the algorithm

The three major sources of approximation errors are the calculation of the PDF in Sect. 3.1, the evaluation of the sums with MC or QMC in Sect. 3.3, and the time-integration of the differential-summation equation in Sect. 3.2. These errors are analyzed and computable estimates of them are proposed in this section and the growth of the computational work is estimated when the number of trajectories M and the number of quadrature points K increase.

4.1   Error in the marginal PDF

The stochastic variable Ψ_j at t^ν in (10) is 1 with probability $p_0(\mathbf{x}, t^\nu)$ in trajectory j of the chemical system. Then according to the law of the large numbers [2, 12], p_0 is approximated at $\mathbf{x} = \mathbf{x}_k$ by

$$\bar{p}^\nu_{0k} = \frac{1}{M} \sum_{j=1}^{M} \Psi_j.$$

In order to estimate the variance of the PDF at t^ν and $\mathbf{x}_k$ the original trajectory matrix is subdivided in J parts each one of size $\tilde{M}$ and with an approximation $\tilde{p}^{\nu,j}_{0k}$. The variance σ^2_{Mk} in $\tilde{p}^{\nu,j}_{0k}$ is estimated by the sample variance

$$s^2_{Mk} = \frac{1}{J-1} \sum_{j=1}^{J} (\tilde{p}^{\nu,j}_{0k} - \bar{p}^\nu_{0k})^2, \tag{17}$$

where the pooled estimate is given by $\bar{p}^\nu_{0k} = \frac{1}{J} \sum_{j=1}^{J} \tilde{p}^{\nu,j}_{0k}$. The error ϵ_{Mk} in $\bar{p}^\nu_{0k}$ is normally distributed by the central limit theorem, $\mathcal{N}(0, \sigma^2_{Mk}/J)$. If σ^2_Ψ is the variance of Ψ_j, then σ^2_{Mk} is $\sigma^2_\Psi/\tilde{M}$ and the variance of $\bar{p}^\nu_{0k}$ is $\sigma^2_\Psi/(J\tilde{M}) = \sigma^2_\Psi/M$. An approximate 95 % confidence interval for ϵ_{Mk} is given by the Student t-distribution [12]

$$|\epsilon_{Mk}| \leq 2s_{Mk}/\sqrt{J}. \tag{18}$$

4.2   Error in the expected values

Suppose that the PDF in (16) is known by its exact value. The sum is the expected value of $\boldsymbol{\omega}$ at time t with the parameter $\boldsymbol{\phi}$. In an MC method, the sum is computed at t^ν by determining trial vectors $\mathbf{x}_k$ with uniformly distributed

components. Let the sum of the i:th component at t^ν be evaluated Q times using K trial vectors each time

$$S_{qi} = \frac{\zeta}{K} \sum_{k=1}^{K} \omega_i(\mathbf{x}^\nu_{K(q-1)+k}, \boldsymbol{\phi}^\nu, t^\nu) \, p_0(\mathbf{x}^\nu_{K(q-1)+k}, t^\nu), \ q = 1, \ldots, Q. \tag{19}$$

Usually, Q is much smaller than K. The pooled estimate of $E[\boldsymbol{\omega}(\mathbf{X}, \boldsymbol{\phi}, t)]$ is

$$S_i = \frac{1}{Q} \sum_{q=1}^{Q} S_{qi}. \tag{20}$$

It approximates the expected value with the error ϵ_{Ki}. The sample variance s_i^2 of S_{qi} is

$$s_i^2 = \frac{1}{Q-1} \sum_{q=1}^{Q} (S_{qi} - S_i)^2. \tag{21}$$

It follows again from the central limit theorem that the error in S_i with probability 0.95 is bounded by

$$|\epsilon_{Ki}| \leq 2s_i/\sqrt{Q}. \tag{22}$$

The error ϵ_{Ki} is normally distributed, $\mathcal{N}(0, \sigma_{Ki}^2/Q)$, where σ_{Ki}^2 is the variance of S_{qi}. According to the central limit theorem, σ_{Ki} decays as $K^{-1/2}$ when K increases. Consequently, the bound on ϵ_{Ki} is proportional to $1/\sqrt{KQ}$.

The error in the QMC method is estimated by scrambling the sequences as in [16, 26]. The digits of a quasirandom number in a base are permuted randomly to obtain the scrambled number. The advantage with scrambling is that the rapid convergence of QMC is preserved while allowing for error estimates as in MC. The sum S_{qi} is computed as in (19). The error estimate is given by (21). Since $\sigma_{Ki} \sim K^{-1}$, the right hand side in (21) decreases as $1/(K\sqrt{Q})$ for increasing K and Q and a moderate m.

4.3 The total summation error

With $\boldsymbol{\omega}_k^\nu = \boldsymbol{\omega}(\mathbf{x}_k^\nu, \boldsymbol{\phi}^\nu, t^\nu)$, we derive the total error in the summation from (16)

$$\begin{aligned} E[\boldsymbol{\omega}(\mathbf{X}_0, \boldsymbol{\phi}, t^\nu)] \ &= \frac{\zeta}{K} \sum_{k=1}^{K} \boldsymbol{\omega}_k^\nu \, p_0(\mathbf{x}_k^\nu, t^\nu) + \boldsymbol{\epsilon}_K \\ &= \frac{\zeta}{K} \sum_{k=1}^{K} \boldsymbol{\omega}_k^\nu \, \bar{p}_{0k}^\nu + \frac{\zeta}{K} \sum_{k=1}^{K} \boldsymbol{\omega}_k^\nu \, (p_0(\mathbf{x}_k^\nu, t^\nu) - \bar{p}_{0k}^\nu) + \boldsymbol{\epsilon}_K. \end{aligned} \tag{23}$$

The second sum on the right hand side is a sum of independent and normally distributed variables $\epsilon_{Mk} = p_0(\mathbf{x}_k^\nu, t^\nu) - \bar{p}_{0k}^\nu$. The difference ϵ_i^ν between the expected value and the sum $\mathbf{S}$ is in the i:th component

$$\epsilon_i^\nu = E[\omega_i(\mathbf{X}_0, \boldsymbol{\phi}, t^\nu)] - S_i = \epsilon_{Ki} + \frac{\zeta}{K}\sum_{k=1}^K \omega_{ki}^\nu \epsilon_{Mk}. \tag{24}$$

It follows from Sect. 4.1 that the sum Σ_i in (24) has the distribution

$$\Sigma_i \sim \mathcal{N}\Big(0, \big(\frac{\zeta}{K}\big)^2 \sum_{k=1}^K (\omega_{ki}^\nu)^2 \frac{\sigma_{Mk}^2}{J}\Big).$$

If $\sigma_{Mk} \leq \sigma_M$ for all k, then the variance of Σ_i is bounded by

$$Var(\Sigma_i) \leq \frac{\zeta}{K}\frac{\sigma_M^2}{J}\frac{\zeta}{K}\sum_{k=1}^K (\omega_{ki}^\nu)^2. \tag{25}$$

By (24), Sect. 4.2, (25), and the bound $\sigma_{Ki} \leq \sigma_K$, $i = 1, \ldots, n$, the total summation error ϵ_i^ν is normally distributed with zero mean value and a variance that is bounded by

$$Var(\epsilon_i^\nu) \leq \frac{\sigma_K^2}{Q} + \frac{\zeta}{K}\frac{\sigma_M^2}{J}\frac{\zeta}{K}\sum_{k=1}^K (\omega_{ki}^\nu)^2, \ i = 1, \ldots, n. \tag{26}$$

The variance σ_K^2 and the upper bound σ_M^2 in (26) are estimated by the sample variances in (17) and (21). Since σ_K^2 decays with increasing K and σ_M^2 with increasing M, $Var(\epsilon_i^\nu)$ can be made as small as we wish by choosing K and M sufficiently large.

4.4   The integration error

The evaluation of $\mathbf{F}$ in (13) is perturbed by the error $\boldsymbol{\epsilon}^{k+1}$ caused by the summation. It is equal to $\boldsymbol{\epsilon}^\nu$ in (24), where t^ν is the time for the most advanced trajectory in time in the SSA but still $t^\nu \leq t^{k+1}$. The discretization error in the approximation of the time derivative is denoted by $\boldsymbol{\tau}^{k+1}$. Following [13] and [23], the local error in the corrector solution $\boldsymbol{\phi}^{k+1}$ of (13) is

$$\boldsymbol{\phi}(t^{k+1}) - \boldsymbol{\phi}^{k+1} = \alpha_0^{-1}\Delta t^k(\boldsymbol{\tau}_c^{k+1} - \boldsymbol{\epsilon}^{k+1}),$$
$$\boldsymbol{\tau}_c^{k+1} = -\frac{1+\theta^k}{6\theta^k}(\Delta t^k)^2 \boldsymbol{\phi}''' + \mathcal{O}((\Delta t)^3), \tag{27}$$

where $\Delta t = \max(\Delta t^k, \Delta t^{k-1})$. The local error in the predictor $\hat{\boldsymbol{\phi}}^{k+1}$ from (14) is

$$\boldsymbol{\phi}(t^{k+1}) - \hat{\boldsymbol{\phi}}^{k+1} = \hat{\alpha}_0^{-1}\Delta t^k(\boldsymbol{\tau}_p^{k+1} - \boldsymbol{\epsilon}^k),$$
$$\boldsymbol{\tau}_p^{k+1} = \frac{1}{6\theta^k}(\Delta t^k)^2 \boldsymbol{\phi}''' + \mathcal{O}((\Delta t)^3). \tag{28}$$

The error equation approximately satisfied by the global error $\mathbf{e}$ in the corrector is

$$\frac{d\mathbf{e}}{dt} - \frac{\partial \mathbf{F}}{\partial \boldsymbol{\phi}}\mathbf{e} = \boldsymbol{\tau}_c - \boldsymbol{\epsilon}. \tag{29}$$

The aim is to control the driving right hand side in (29) by taking Δt^k sufficiently small and K and M sufficiently large.

The leading term in the temporal discretization error $\boldsymbol{\tau}_c^{k+1}$ is computed by first combining (27) and (28) and then obtaining an estimate of $\boldsymbol{\phi}'''(t^{k+1})$. This estimate is inserted into $\boldsymbol{\tau}_c^{k+1}$ in (27) to arrive at an expression for the j:th entry

$$\tau_{cj}^{k+1} - \epsilon_j^{k+1} = -\frac{1 + 2\theta^k}{2 + 3\theta^k}\left(\frac{\phi_j^{k+1} - \hat{\phi}_j^{k+1}}{\Delta t^k} + (1 + \theta^k)\epsilon_j^k + \epsilon_j^{k+1}\right) + \mathcal{O}(\Delta t^3). \tag{30}$$

Let the variance of ϵ_j^l be denoted by σ_ϵ^2. Since ϵ_j^l is normally distributed with mean 0, $\mathcal{N}(0, \sigma_\epsilon^2)$, (see Sect. 4.3) and independent of ϵ_j^ℓ, $\ell \neq l$, the leading term in (30) is

$$\mathcal{N}(-\frac{1 + 2\theta^k}{2 + 3\theta^k}\frac{\phi_j^{k+1} - \hat{\phi}_j^{k+1}}{\Delta t^k}, \left(\frac{1 + 2\theta^k}{2 + 3\theta^k}\right)^2((1 + \theta^k)^2 + 1)\sigma_\epsilon^2). \tag{31}$$

The first term in (30) depends on Δt^2 but the second and third terms are independent of Δt. Suppose that we want the error in the right hand side of (29) to be less than η with a 95 % confidence interval. Choose a fraction γ, $0 < \gamma < 1$, a time step Δt^k, K, Q, M, and J such that

$$|\frac{\phi_j^{k+1} - \hat{\phi}_j^{k+1}}{\Delta t^k}| \leq \frac{2 + 3\theta^k}{1 + 2\theta^k}\gamma\eta,$$

$$\sigma_\epsilon \leq \left(\frac{\sigma_K^2}{Q} + \frac{\zeta}{K}\frac{\sigma_M^2}{J}\frac{\zeta}{K}\sum_{k=1}^{K}(\omega_{kj}^\nu)^2\right)^{1/2} \leq \frac{2 + 3\theta^k}{1.96(1 + 2\theta^k)\sqrt{1 + (1 + \theta^k)^2}}(1 - \gamma)\eta, \tag{32}$$

for all j using (26). Then by (30) we have

$$|\tau_{cj}^{k+1} - \epsilon_j^{k+1}| \leq \eta \tag{33}$$

with probability 0.95. The variance σ_{Ki}^2 and the bound σ_M are estimated by (21) and (17). The parameter γ determines the fraction of the bound on $\tau_{cj}^{k+1} - \epsilon_j^{k+1}$ to be satisfied by the first, deterministic term in the error. With constant time steps, $\theta^k = 1$, the upper bounds in (32) are simplified to

$$|\frac{\phi_j^{k+1} - \hat{\phi}_j^{k+1}}{\Delta t^k}| \leq 1.67\gamma\eta, \quad \sigma_\epsilon \leq 0.38(1 - \gamma)\eta.$$

4.5 Computational work

For large systems and a large number of trajectories M, the contribution of SSA will dominate the total computational work of the hybrid solver. The work of SSA grows linearly with ρ (and also with m) in (8) for one trajectory.

Suppose that the work in SSA to determine M trajectories is $c\rho M$ and that the proportion ϑ of the total CPU time in the hybrid algorithm is spent by SSA. Assuming c to be constant, the quotient between the hybrid time T_{HYB} and the time T_{SSA} for SSA applied to all variables is

$$\frac{T_{SSA}}{T_{HYB}} = \frac{cRM}{c\rho M} = \frac{\vartheta R}{\rho}. \tag{34}$$

With ϑ close to 1, we can expect the reduction to be of the order ρ/R.

The overhead associated with the solution of the deterministic equations (7) can be estimated by considering the other major contributions to the algorithm. Assuming that a total of QK QMC vectors are used in the evaluation of the sum, p_0 has to be calculated at QK points. For the evaluation of p_0, the trajectory matrix $\mathbf{T}$ is first sorted according to rows, using the MATLAB routine `sortrows`, which is an implementation of the quicksort algorithm with a work of $\mathcal{O}(mM \log M)$. Obviously, many rows of $\mathbf{T}$ will consist of equal vectors for M reasonably large. They are removed using a modification of the MATLAB built-in routine `unique`, with a work of $\mathcal{O}(mM)$, giving a modified trajectory matrix $\hat{\mathbf{T}}$ with $\hat{M} \leq M$ rows. One quadrature point is then found in $\hat{\mathbf{T}}$ using binary search with work proportional to $m \log \hat{M}$. In total, the work of evaluating the sum is $\mathcal{O}(mM \log M) + \mathcal{O}(mQK \log \hat{M})$. The evaluation of the full Jacobian $\mathbf{J}$ and $\mathbf{F}$ would need work of $\mathcal{O}(n^2 QK)$ and $\mathcal{O}(nQK)$, respectively, and the solution of the system of linear equations in the Newton iterations costs $\mathcal{O}(n^2)$ operations with a factorized $\mathbf{J}$. Thanks to the sparsity of the Jacobian the work is typically much lower, e.g. with an iterative method. The major terms in descending order in the estimate of the computational work W in one time step of the hybrid algorithm are

$$W = cmM + c_1 mM \log M + c_2 mQK \log \hat{M}. \tag{35}$$

The first term in (35) due to SSA is usually much larger than the other ones in the experiments in Sect. 5, i.e. ϑ is close to 1.
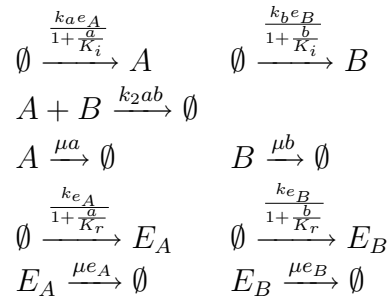
5 Numerical results

The hybrid algorithm is applied to the simulation of three different chemical systems. The first system is small with two metabolites and two enzymes. The production of the metabolites is controlled by the enzymes. The second system

models a molecular clock where the reaction rate equations fail to produce oscillations for a certain parameter value. Introducing a few stochastic variables makes the model much more robust to changes in the parameter. This is an example where addition of stochastic noise improves the original model based on the reaction rate equations. The third system is a mitogen-activated protein kinase (MAPK) signaling cascade with 22 molecular species. Only two of these species are treated stochastically. The results from the simulations with the full system and the reduced system are similar but the computational work to simulate the reduced system with good accuracy is much lower.

The most time consuming parts of the hybrid solver, i.e. SSA, evaluation of p_0, MC or QMC summation, and computation of the Jacobian $\mathbf{J}$, are implemented in C and wrapped as mex-files in a MATLAB environment. Different systems are defined by the implementation of one function that performs SSA on the reduced system, and one function that returns the right hand side of the differential-summation equations. To facilitate the generation of different hybrid splittings, a graphical tool has been developed in Python that reads models in the SBML format [17] and automatically generates the necessary C files given a desired partitioning. The reaction rate equations are integrated by MATLAB's `ode15s`. The code for generation of the quasi-sequence is written as a mex-file calling a Fortran subroutine from [16]. This part is executed once initially and contributes little to the total CPU time if the number of time steps is sufficiently large.

5.1   Metabolites controled by enzymes

The first system is a simple generic model with two metabolites A and B and two enzymes E_A and E_B as in [22]. The production of A and B is regulated by the enzymes. The reactions are

$$\emptyset \xrightarrow{\frac{k_a e_A}{1+\frac{a}{K_i}}} A \qquad \emptyset \xrightarrow{\frac{k_b e_B}{1+\frac{b}{K_i}}} B$$

$$A + B \xrightarrow{k_2 ab} \emptyset$$

$$A \xrightarrow{\mu a} \emptyset \qquad B \xrightarrow{\mu b} \emptyset$$

$$\emptyset \xrightarrow{\frac{k_{eA}}{1+\frac{a}{K_r}}} E_A \qquad \emptyset \xrightarrow{\frac{k_{eB}}{1+\frac{b}{K_r}}} E_B$$

$$E_A \xrightarrow{\mu e_A} \emptyset \qquad E_B \xrightarrow{\mu e_B} \emptyset$$

The reaction constants in (5.1) are:

k_a	k_b	k_2	K_i	μ	k_{eA}	k_{eB}	K_r
0.3	0.3	0.001	60	0.002	0.02	0.02	30

Table 1: The parameters for the metabolite-enzyme model.

The variables are partitioned so that the metabolites A and B are treated as stochastic variables and the enzymes E_A and E_B are assumed to be normally distributed with a small variance. In this case, the number of stochastic variables decreases from $N = 4$ in the full model to $m = 2$ and $n = 2$, the number of reactions is reduced from nine to $\rho = 5$ in (8), and the number of reactions with propensities depending on $\mathbf{x}$, κ, is seven in (9).
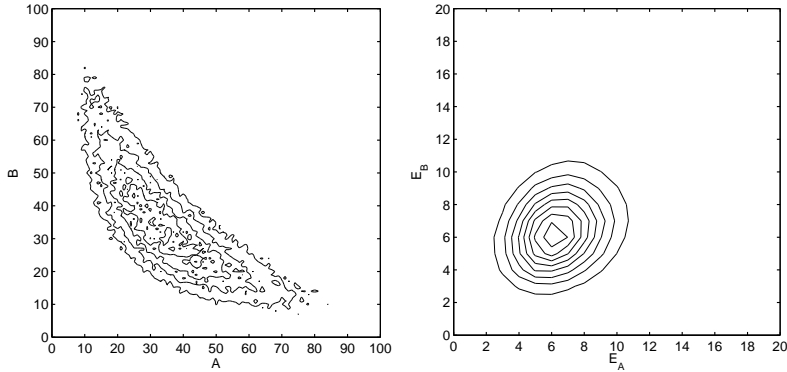

Fig. 1: Isolines of the marginal probability density at $t = 1000$ for A and B (left) and E_A and E_B (right) computed by SSA with $M = 10^5$.

The marginal probabilities from the SSA for the metabolites and the enzymes are plotted in Fig. 1. The number of trajectories M approximating the PDFs is 10^5. The system is close to the steady state at $t = 1000$. The fluctuations of the species A and B in (10) appear to be larger than for the enzymes in the figure.

Different summation techniques for (16) are evaluated in Fig. 2 with the PDF based on $M = 10^5$ realizations at $t = 1000$. The slope of the curves has been determined by a least squares fit to the data. The sums $\mathbf{S}_q, q = 1, \ldots, Q$, are computed for $Q = 30$. Then the sample or empirical standard deviation is determined (componentwise) as in (21).

In Fig. 2, the first component of the sample standard deviation is displayed. Raw MC summation with pseudorandom numbers is compared to raw QMC summation using numbers from the Faure sequence generated with the algorithm in [16]. The acceptance-rejection method (A-R) [2, 12] and a smoothing alternative [2, 25] are tested. The discontinuity in the A-R method is replaced by a linear function between 0 and 1 in [25]. One important consideration is the fact that when for example the A-R method is used to generate numbers from the distribution, fewer evaluation points will be accepted and used than in e.g. the raw MC method. This means that K in (16) will be different for different methods. Thus, the errors reported in Fig. 2 refer to the error obtained with a fixed number of trial points K. A fair comparison with the acceptance-rejection methods would require sampling until the number of accepted points were within one unit of K, but what we are interested in is which method gives the smallest error per

16

generated random number, to account for the cost in generating the numbers and most of all in evaluating the PDF. It has to be evaluated for all trial points, not just the accepted ones. The best convergence rate with the lowest error is achieved by standard QMC. This is the preferred method in the sequel.
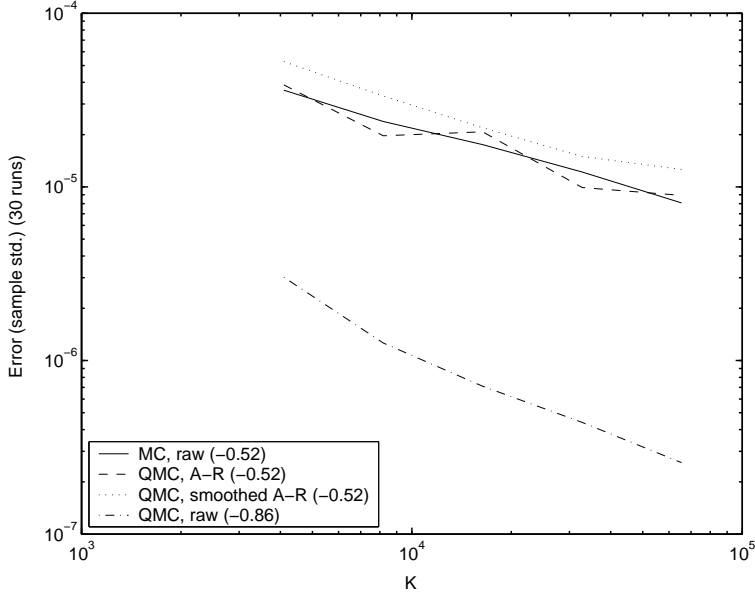

Fig. 2: The sample standard deviation for different MC and QMC strategies. The number in parenthesis is the inclination of the curve.

The system is simulated with the hybrid solver. The time steps are chosen adaptively with a relative error tolerance of 0.01. The maximal time step Δt_{max} is 5. The summation is performed by using a maximum of $Q = 10$ scrambled sequences. They are generated and stored initially and then used to form estimates of the sum and the error as in (20) and (22). In the simulations reported here, $K = 2^{15} \approx 33 \cdot 10^3$ quasirandom points were used for each sequence. If the tolerance in (32) was met for any number of sequences less than Q, no more sequences were used. The maximal value of the estimated leading terms of the error in (30) and σ_ϵ in (32) are shown in Fig. 3. For this system, the error tolerance ($\eta = 0.01$, $\gamma = 0.9$) was easily met, and the time step reached its maximal value after approximatively 55 s and remained fixed for the rest of the simulation.
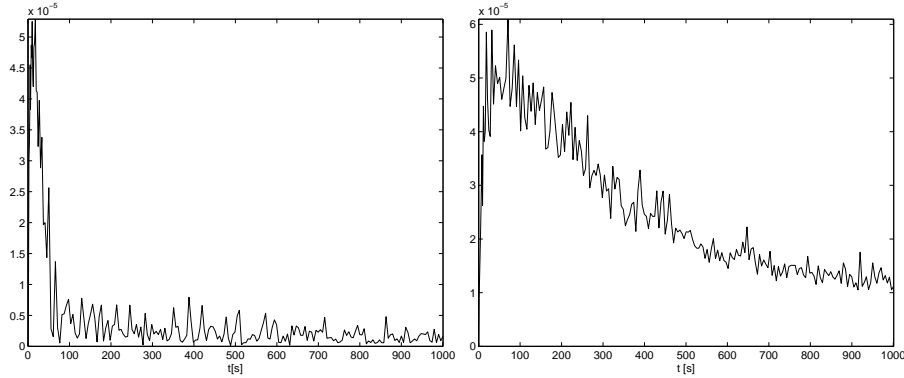
17

Fig. 3: The maximal local time discretization errors (left) and the maximal standard deviation in the QMC approximation of the sum (right) with $K = 2^{15}, M = 10^5$.

From (32) and Sect. 4.2 some conclusions can be drawn concerning how to choose K and Q. The bound of σ_ϵ^2 in (32) consists of two terms, one corresponding to σ_K^2 and the other to σ_M^2. Fig. 4 shows these terms separately as computed during the simulation. In this case, the second term dominates.
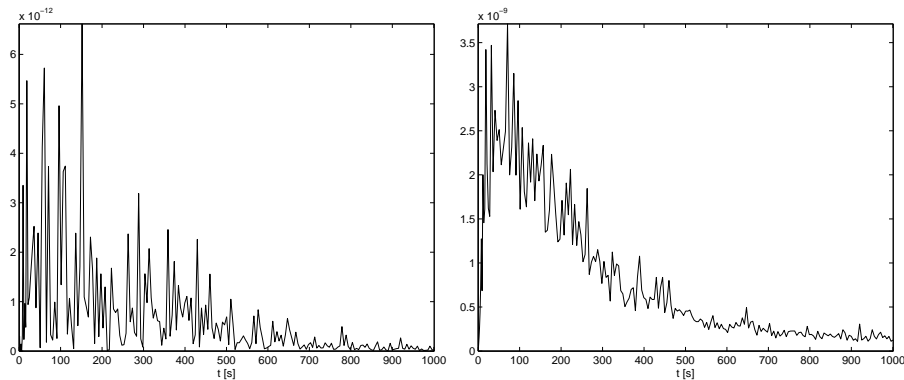

Fig. 4: The first term involving σ_K^2 (left) and the second term involving σ_M^2 (right) in (32) with $K = 2^{15}, M = 10^5$.

The state space is not large in this example. An upper bound on the second term $\sigma_{\epsilon 2}^2$ in the error is

$$\sigma_{\epsilon 2}^2 = \frac{\zeta}{K} \frac{\sigma_M^2}{J} \frac{\zeta}{K} \sum_{k=1}^{K} (\omega_{kj}^\nu)^2 \leq \frac{\zeta^2 \sigma_M^2}{JK} \max_{\mathbf{x} \in \Omega} \omega_i(\mathbf{x}, \phi, t)^2 \tag{36}$$

where ω_i consists of the reaction propensities. In a model of a biological reactor, the maximum of these propensities is seldom very large. Furthermore, in a reaction rate model ω_i corresponds to the time derivative of the species i, that is

$$|\omega_i(\phi, t)| = |\frac{d\phi_i}{dt}|.$$

18

For systems suitable for a hybrid approach, ideally we are allowed to take large time steps in the deterministic part of the solver. For such systems the solution will vary slowly, and the absolute values of the derivatives will be small. One example where this is true is found to the right in Fig. 5 below. Obviously, for stiff systems, the absolute value of the derivative will be large in some time intervals. An example where this happens will be studied in the next section.

In either case, it is reasonable to conclude that for larger systems, ζ will dominate the nominator of $\sigma_{\epsilon2}^2$ and it grows rapidly with an increasing number of stochastic variables. To ensure that this term stays small it is advantageous to choose large values of K and M. Also, the error in the QMC approximation of the sum decreases as $1/(K\sqrt{Q})$. This taken together, suggests that for a constant KQ the smallest error is obtained for $Q = 1$, i.e. one single, long QMC sequence. The price is that no simple error estimation is available. The results of the other two simulations in Sects. 5.2 and 5.3 are consistent with these conclusions.

The hybrid method captures the behavior of the expected values of the full model but the marginal PDF for A and B in Fig. 5 is not extended along the diagonal in the same manner as the SSA solution in Fig. 1. However, the distribution obtained with this solver compares very well with the results in [22] where a Fokker-Planck equation approximates the master equation.
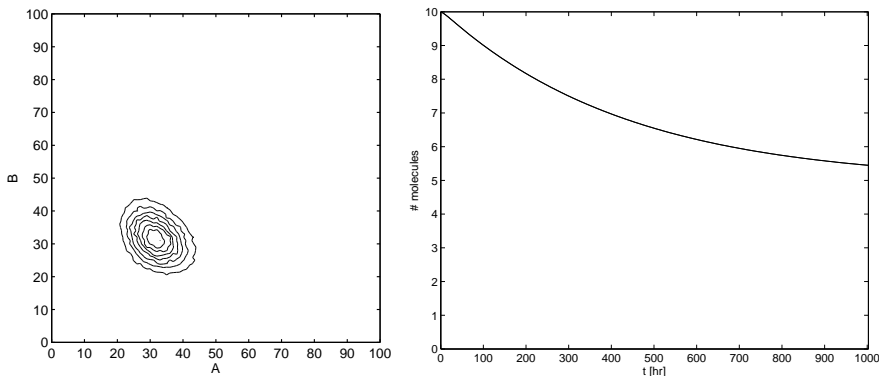

Fig. 5: The probability density for A and B at $t = 1000$ (left) and the expected values for E_A and E_B (right) computed with the hybrid solver.

The MATLAB tool 'Profiler' has been invoked to determine the time spent in different parts of the algorithm. The number of quasirandom points for the summation is $2^{15} \approx 33 \cdot 10^3$ and the final time is $t = 103$ in Table 2.

19

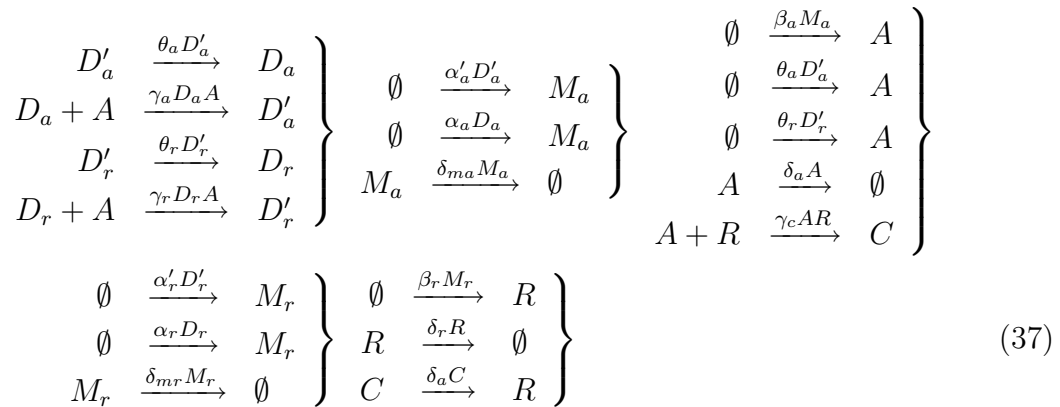| Number of trajectories	10^3	10^4	10^5	10^6
Time spent in SSA [%]	3.3	18.7	44.7	50.9
Total CPU time [s]	35	48	181	1615

Table 2: Time spent in SSA.

We see in Table 2 that for many trajectories the SSA requires about 50 % of the total time. The other major computational cost is the sorting of the state vectors with algorithms provided by MATLAB. They require almost 40 % of the execution time for 10^6 trajectories. For this particular problem, there is no gain in time using the hybrid scheme compared to the SSA. Indeed, for 10^6 trajectories the time spent in SSA in the hybrid solver was 821 s compared to 872 s for the full system.

5.2 Circadian rhythm model

The oscillator [1, 34] is a model for circadian rhythms, illustrating some common control components that have been observed in such systems. This kind of control system is designed to assure periodic oscillations of certain molecular species in order to establish a circadian rhythm in the organism. Obviously, a system of this kind is very complicated in an organism, and this model system is simplified.

The model has nine molecular species. Two genes D_a and D_r and their corresponding mRNA, M_a and M_r, are controlled by an activator and a repressor A and R, synthesized from the respective mRNA. Furthermore, the activator and repressor can associate and form a complex C, in which the activator A is degraded. The variables D'_a and D'_r are the genes D_a and D_r with a bound activator. In the model it is assumed that there is only one gene coding for the repressor and activator. Thus $D_a + D'_a = 1$, and the same is true for the repressor gene. The 18 reactions for the nine molecular species are

$$
\left.
\begin{aligned}
D'_a & \xrightarrow{\theta_a D'_a} D_a \\
D_a + A & \xrightarrow{\gamma_a D_a A} D'_a \\
D'_r & \xrightarrow{\theta_r D'_r} D_r \\
D_r + A & \xrightarrow{\gamma_r D_r A} D'_r
\end{aligned}
\right\}
\quad
\left.
\begin{aligned}
\emptyset & \xrightarrow{\alpha'_a D'_a} M_a \\
\emptyset & \xrightarrow{\alpha_a D_a} M_a \\
M_a & \xrightarrow{\delta_{ma} M_a} \emptyset
\end{aligned}
\right\}
\quad
\left.
\begin{aligned}
\emptyset & \xrightarrow{\beta_a M_a} A \\
\emptyset & \xrightarrow{\theta_a D'_a} A \\
\emptyset & \xrightarrow{\theta_r D'_r} A \\
A & \xrightarrow{\delta_a A} \emptyset \\
A + R & \xrightarrow{\gamma_c A R} C
\end{aligned}
\right\}
$$

$$
\left.
\begin{aligned}
\emptyset & \xrightarrow{\alpha'_r D'_r} M_r \\
\emptyset & \xrightarrow{\alpha_r D_r} M_r \\
M_r & \xrightarrow{\delta_{mr} M_r} \emptyset
\end{aligned}
\right\}
\quad
\left.
\begin{aligned}
\emptyset & \xrightarrow{\beta_r M_r} R \\
R & \xrightarrow{\delta_r R} \emptyset \\
C & \xrightarrow{\delta_a C} R
\end{aligned}
\right\}
\tag{37}
$$

The reaction constants are found in Table 3. For these parameters the system exhibits a limit cycle but if the parameter δ_r is sufficiently small then the macroscopic reaction rate equations quickly reach a stable fixed point and the oscillations stop, see Fig. 6. It is shown by Vilar et al in [34] that a mesoscopic description of the system continues to produce reliable oscillations. The stochastic noise is obviously sufficient to perturb the trajectories away from the fixed point to initiate new cycles. The parameter δ_r has two different values 0.2 and 0.08 in the numerical experiments.

α_A	α_a'	α_r	α_r'	β_a	β_r	δ_{ma}	δ_{mr}
50.0	500.0	0.01	50	50.0	5.0	10.0	0.5

δ_a	δ_r	γ_a	γ_r	γ_c	Θ_a	Θ_r
1.0	0.2	1.0	1.0	2.0	50.0	100.0

Table 3: Parameters for the Vilar oscillator.
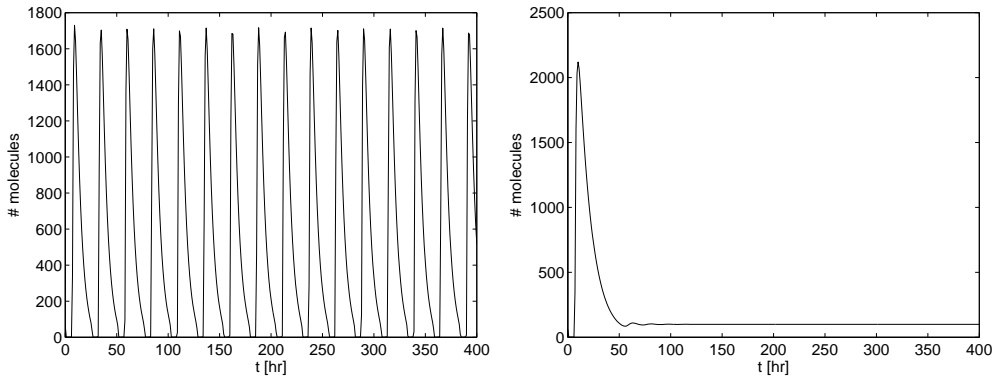
Fig. 6: Time evolution of the repressor R computed with the reaction rate equations. The parameter δ_r is 0.2 (left) and 0.08 (right).

The variables are partitioned in the hybrid method into two subsets: A, R and C are treated stochastically, corresponding to $\mathbf{X}$ in Section 2 with $m = 3$, while the other variables are treated deterministically, corresponding to $\mathbf{Y}$ with $n = 6$. There are ten stochastic reactions ($\rho = 10$ in (8)). The initial conditions for the deterministic variables are 0.2 for the genes and 0 for the other species. The stochastic variables are initiated as normal distributions centered around 10. The probability density p_0 is approximated with $M = 10^5$ trajectories in (10) and $K = 2^{18} \approx 2.62 \cdot 10^5$ quasi-random points in (16) are used in each sequence in the integration algorithm ($Q = 5$). The system was solved with adaptive time step selection with the error tolerance 0.05.
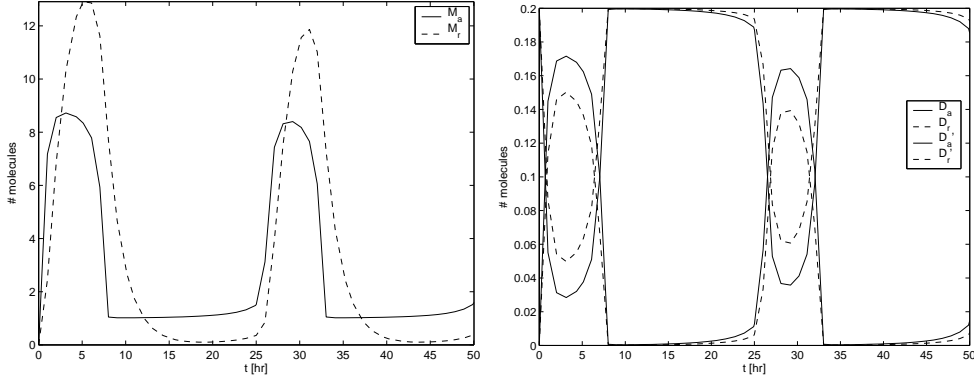
21

Fig. 7: Time evolution of the mRNA variables M_a and M_r (left) and the gene variables D_a, D'_a, D_r, and D'_r (right) computed with the hybrid method when $\delta_r = 0.2$.

Sustained oscillations are obtained with the hybrid algorithm for $\delta_r = 0.2$ in Figs. 7 and 8. Fig. 8 also displays the length of the time steps taken by the solver. As can be seen, small time steps are needed in regions where the solution changes rapidly (cf. Fig. 7). Recomputing time steps after a failure to satisfy the error tolerance is necessary occasionally in this example. This incurs an extra cost when this oscillator is simulated with the hybrid algorithm. On the other hand, sometimes a small time discretization error is not the highest priority since the understanding of concepts is often the major goal rather than exact predictions.
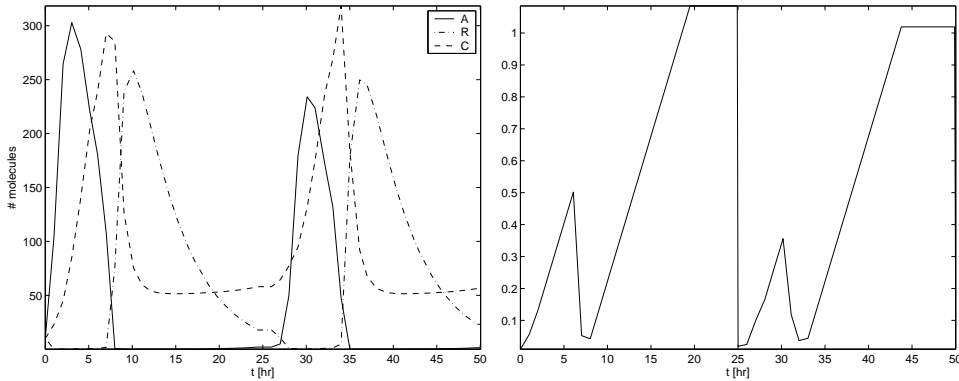
Fig. 8: Time evolution of the expected values of the activator A, the repressor R, and the complex C computed with the hybrid method when $\delta_r = 0.2$ (left) and the time steps Δt taken in the adaptive solver (right).

Both the maximal standard deviation of the QMC summation σ_K and the standard deviation of the PDF measured by σ_M in (32) exhibit periodic oscillations with distinct maxima and minima in Fig. 9. The maxima of σ_K correspond to time intervals with rapid variation of the solution, and the minima of σ_M correspond to time intervals where the mean values of the stochastic variables take small values and ζ is small. Unfortunately, while the error due to the discrete

time steps was controlled easily, the error tolerance dictated by σ_ϵ in (32) could not be fulfilled in all time intervals. This was mostly due to the second term involving σ_M. In the regions with a fast change in the solution the value of the sum becomes large, combined with a relatively large state space. This problem is ameliorated by using one single QMC sequence of $5 \cdot 2^{18}$ points instead, according to the discussion in the previous section.
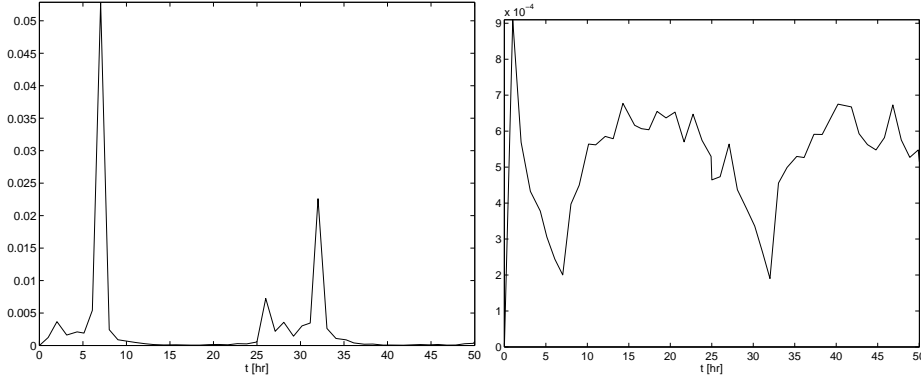

Fig. 9: The maximal standard deviation of the QMC summation, σ_K, (left) and the standard deviation in the PDF (right) for $\delta_r = 0.2$.

For $\delta_r = 0.08$ where the reaction rate equations reach a steady state, see Fig. 6, while the solution computed with the hybrid solver continues to produce reliable oscillations in agreement with the numerical experiments in [5, 7] and the conclusions in [34]. However, with the splitting used in the above example, small time steps are needed in order to ensure oscillations. The phase of the oscillations also depends on the time discretization error. This is not the only possible partitioning that give these results. Fig. 10 depicts the solution when two different partitionings of the system have been considered. For both solutions, fixed time steps were used. To the left we have the result of the original splitting when $\Delta t = 10^{-3}$ and one single QMC sequence ($Q = 1$) of $2^{17} \approx 1.31 \cdot 10^5$ points was used. To the right we have the solution when only A and R are treated stochastically. Here, $M = 1.0 \cdot 10^4$, $\Delta t = 0.1$, $Q = 1$, and $K = 2^{17}$. In both cases, oscillations occur, but the latter configuration is obviously far less computationally demanding, and benefits more from the hybrid splitting.
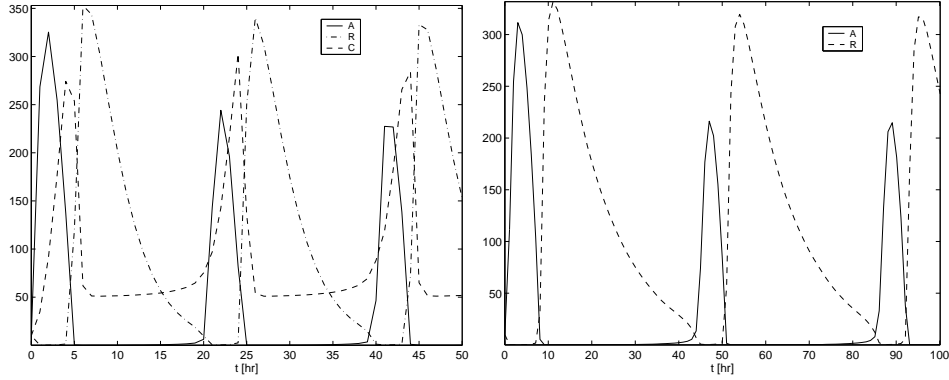
Fig. 10: Solution for $\delta_r = 0.08$ when A, R and C are treated stochastically (left) and the corresponding simulation with only A and R as stochastic variables (right).

The contribution of the SSA algorithm to the total time required to solve the system is studied in Tables 4 and 5. For this system, the stochastic simulations are more demanding than for the previous example in Sect. 5.1. For $M = 10^6$ SSA consumes more than 90 % of the CPU time when $K = 2^{15}$ and the quotient between the CPU time for SSA, T_{SSA}, and the CPU time for the hybrid algorithm, T_{HYB}, is greater than one for $M \geq 10^4$. For the larger value of K in Table 5, relatively more time is spent on the calculations of the sums. This system consists of only nine variables, and for larger systems there is plenty of time to save if a good splitting of the variables can be found.

Number of trajectories	10^3	10^4	10^5	10^6
Time spent in SSA [%]	18.6	53.1	83.9	91.8
Total CPU time in 10^3 s	0.18	0.59	3.50	29.70
T_{SSA}/T_{HYB}	0.43	1.32	2.19	2.58

Table 4: Time spent in SSA when the system was simulated with a fixed time step $\Delta t = 0.5$ to the final time 100 s and $K = 2^{15} \approx 33 \cdot 10^3$.

Number of trajectories	10^3	10^4	10^5	10^6
Time spent in SSA [%]	3.1	14.8	42.4	73.9
Total CPU time in 10^3 s	0.47	0.96	3.31	18.89
T_{SSA}/T_{HYB}	0.04	0.20	0.58	1.02

Table 5: Time spent in SSA when the system was simulated with a fixed time step $\Delta t = 0.5$ to the final time 25 s and $K = 2^{16} \approx 65 \cdot 10^3$.
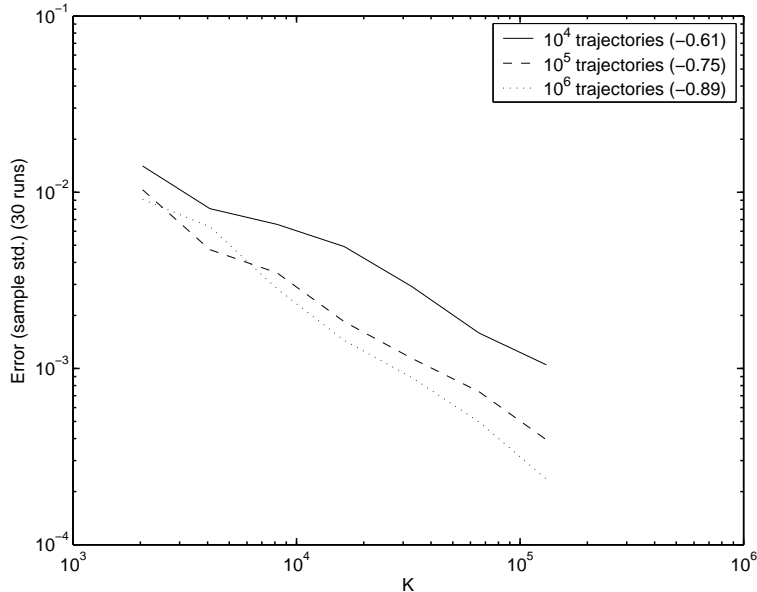
Fig. 11: The absolute summation error for different number of trajectories M and quadrature points K. The number in parenthesis is the inclination of the curve.

The summation error of the QMC quadrature determined by (22) for this problem is displayed in Fig. 11. The sum in (16) with p_0 at $t = 25s$ is computed using the Faure sequences for different number of trajectories M and different number of quadrature points K. The error is based on 30 randomized QMC sequences, i.e $Q = 30$ in (21). As can be seen, the convergence rate and the slope increase with increasing number of trajectories. This is expected, since a larger M implies a smoother p_0 and a smaller error in the quadrature.

5.3   Signaling cascade

As a final example we will consider a model of a mitogen-activated protein kinase (MAPK) signaling cascade [20, 21]. These receptor mediated signal transduction pathways are conserved regulatory systems, and consist of three sequentially acting kinases. Kinases are proteins that modify other proteins by the phosphorylation of certain amino acid residues. This modification has a different effect on different proteins, and could lead to e.g. changes in binding properties to DNA and RNA or to other proteins. In this case, the first protein in the cascade, RAF, induces two phosphorylations of MEK, the second component in the chain. Doubly phosphorylated MEK, MEKpp, in turn induces two phosphorylations of MAPK. The output signal in the scheme, doubly phosphorylated MAPK, MAPKpp, can dimerize and in this form be transported into the nucleus of the cell where it phosphorylates a number of transcription factors, proteins directly involved in the regulation of the transcription of genes to mRNA [21]. The model

in [20] also takes into account the effect of scaffolds but this part of the model is not included here.

The model has 22 variables, the kinases RAF, MEK, and MAPK, the de-phosphatases RAFPH and MEKPH, which remove phosphate groups from their corresponding kinase, and the possible dimers formed in the reactions. These 22 variables take part in 30 reactions.

A deterministic simulation of the system with the reaction rate equations is found in Fig. 12. Only the singly and doubly phosphorylated MAPK are displayed. Except for a few species, the components vary slowly in time. With the parameters given in [21], the average number of MAPKpp molecules at a maximal output signal is below one. A stochastic simulation with SSA reveals that the number of molecules of doubly phosphorylated MAPK vary from zero to a few molecules after an initial time when no active MAPK is present, see Fig. 12.
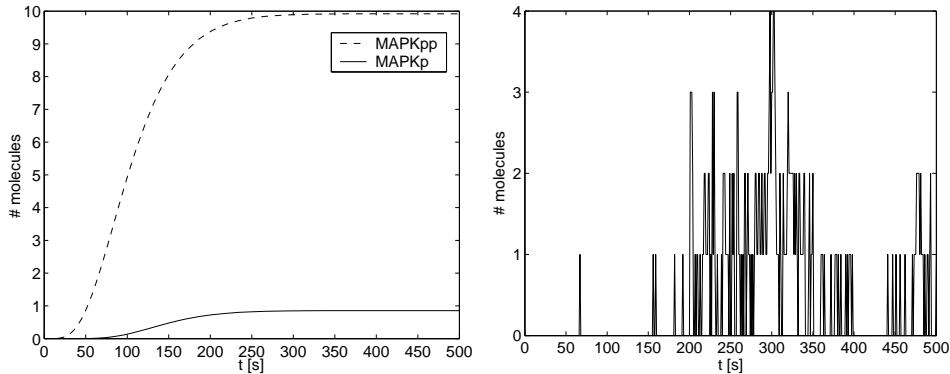

Fig. 12: Simulation of MAPKp and MAPKpp with the reaction rate equations (left) and simulation of MAPKpp in one trajectory with SSA (right).

Suppose that we want to construct a larger model where a control system like this is one component. We might then be interested in the stochastic variation of the output signal in MAPKpp, while the detailed stochastic information concerning the other components is less important. This is a scenario where the hybrid approach provides a speedup compared to the full SSA and more modeling accuracy than the reaction rate equations. To evaluate the performance of the hybrid method, the state space is divided so that singly and doubly phosphorylated MAPK (MAPKp and MAPKpp) are treated as stochastic variables, $\mathbf{X}$ and $m = 2$ in Sect. 2, and the other compounds as deterministic variables, $\mathbf{Y}$ and $n = 20$. In this way, 30 reactions for 22 species are reduced to nine reactions with a stochastic component, $\rho = 9$ in (8), and 20 differential-summation equations.
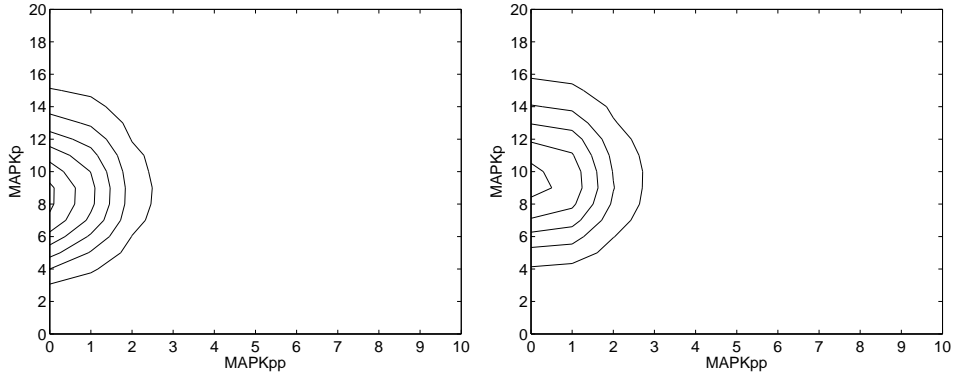
Fig. 13: Isolines of the marginal probability distribution for MAPKp and MAPKpp with SSA (left) and the hybrid method (right).

The isolines for $p_0(\mathbf{x}, 500)$ using SSA for the full system and the hybrid solution are compared in Fig. 13 using $M = 10^5$ trajectories. It is evident in the figure that the hybrid solver is able to capture the stochastic properties of the activated MAPK species. Since there are only two stochastic variables and their copy numbers are small, a relatively small number of quadrature points are needed and the hybrid solver is also much more efficient than SSA for the full system. In Table 6, we find that with $M = 10^6$ SSA is more than seven times slower than the hybrid method. This system is still fairly small. For example, the same model extended to include scaffolds [21] has 89 variables participating in 300 chemical reactions. For a system of this size, the hybrid solver is expected to be a lot faster than SSA if a good separation of the variables can be chosen.

Number of trajectories	10^3	10^4	10^5	10^6
Time spent in SSA [%]	6.4	7.9	37.8	58.1
Total CPU time in 10^3 s	0.45	0.51	1.0	6.68
T_{SSA}/T_{HYB}	0.12	0.93	4.63	7.14

Table 6: Time spent in SSA when the system was simulated with a fixed time step $\Delta t = 1s$ to the final time $t = 200s$ and $K = 2^{15} \approx 33 \cdot 10^3$.

6 Conclusions

From the numerical experiments the conclusion is that the hybrid solver is able to capture important features of the fully mesoscopic models, while keeping the number of stochastically treated variables at a manageable level.

For one of the model systems considered, the reactions are too few and the splitting done in such a way that an improvement in execution time over the full

27

SSA algorithm can not be obtained. It is shown however, that for systems where this splitting achieves a sufficient reduction in the number of reactions and the rate constants involved in those reactions, a considerable speedup compared to a fully stochastic simulation can be obtained. For a larger system, the hybrid solver is shown to execute up to seven times faster than the full SSA with the chosen splitting, while it is still able to retain the stochastic properties of the original model. Alternatively, the hybrid model can be viewed upon as a way of improving the macroscopic model by introducing stochasticity in some components. With this viewpoint, the hybrid solver is more computationally demanding than the ODE models, but gives more realistic results at a low additional cost.

The major bottleneck in the time stepping scheme of the hybrid solver is, apart from the SSA simulation, the evaluation of the probability distribution function $p_0(\mathbf{x}, t)$. The number of evaluation points is determined by the performance of the summation algorithm. It is therefore crucial to develop a scheme that gives a small error with few evaluated quadrature points. Here, the sum is computed with a Quasi Monte Carlo method. Monte Carlo methods are well suited for high dimensional summation. Even if the dimensions are not particularly high for the systems considered here, a generally applicable method is required if the intention is to be able to simulate different systems without extensive reprogramming by the user. One problem with QMC methods is that the higher convergence rate compared to pseudorandom numbers depends on the summand being sufficiently smooth. Since we are dealing with a discrete distribution with stochastic errors this is not always the case and we have seen that for a small quadrature error a large number of trajectories is needed. Even when this requirement can be met, the convergence rate is still not as high as in the ideal smooth case.

Acknowledgment

References

[1] N. BARKAI, S. LEIBLER, Circadian clocks limited by noise, *Nature*, 403 (2000), p. 267–268.

[2] R. E. CAFLISCH, Monte Carlo and quasi-Monte Carlo methods, *Acta Numerica*, 1998, p. 1–49.

[3] Y. CAO, D. GILLESPIE, L. PETZOLD, Multiscale stochastic simulation algorithm with stochastic partial equilibrium assumption for chemically reacting systems, *J. Comput. Phys.*, 206 (2005), p. 395–411.

[4] J. ELF, J. PAULSSON, O. G. BERG, M. EHRENBERG, Near-critical phenomena in intracellular metabolite pools, *Biophys. J.*, 84 (2003), p. 154–170.

[5] S. ENGBLOM, Computing the moments of high dimensional solutions of the master equation, Technical report 2005-020, Dept of Information Technology, Uppsala University, Uppsala, Sweden, 2005, to appear in *Appl. Math. Comput.*, available at `http://www.it.uu.se/research/reports/2005-020/`.

[6] H. FAURE, Discrépance de suites associées à un système de numération (en dimension *s*), *Acta Aritm.*, 41 (1982), p. 337–351.

[7] L. FERM, P. LÖTSTEDT, Numerical method for coupling the macro and meso scales in stochastic chemical kinetics, Technical report 2006-001, Dept of Information Technology, Uppsala University, Uppsala, Sweden, 2006, available at `http://www.it.uu.se/research/publications/reports/2006-001/`.

[8] L. FERM, P. LÖTSTEDT, P. SJÖBERG, Conservative solution of the Fokker-Planck equation for stochastic chemical reactions, Technical report 2004-054, Dept of Information Technology, Uppsala University, Uppsala, Sweden, 2004, to appear in *BIT*, available at `http://www.it.uu.se/research/publications/reports/2004-054/`.

[9] C. W. GARDINER, *Handbook of Stochastic Methods*, 2nd ed., Springer, Berlin, 2002.

[10] D. T. GILLESPIE, A general method for numerically simulating the stochastic time evolution of coupled chemical reactions, *J. Comput. Phys.*, 22 (1976), p. 403–434.

[11] D. T. GILLESPIE, Approximate accelerated stochastic simulation of chemically reacting systems, *J. Chem. Phys.*, 115 (2001), p. 1716–1733.

[12] P. GLASSERMAN, *Monte Carlo Methods in Financial Engineering*, Springer, New York.

[13] E. HAIRER, S. P. NØRSETT, G. WANNER, *Solving Ordinary Differential Equations I, Nonstiff Problems*, 2nd ed., Springer-Verlag, Berlin, 1993.

[14] E. HASELTINE, J. RAWLINGS, Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics, *J. Chem. Phys.*, 117 (2002), p. 6959–6969.

[15] M. HEGLAND, C. BURDEN, L. SANTOSO, S. MACNAMARA, H. BOOTH, A solver for the stochastic master equation applied to gene regulatory networks, *J. Comput. Appl. Math.*, to appear.

[16] H. S. Hong, F. J. Hickernell, Algorithm 823: Implementing scrambled digital sequences, *ACM Trans. Math. Softw.*, 29 (2003), p. 95–109.

[17] M. Hucka, A. Finney, A. Sauro, H. M. Bolouri, H. Doyle, J. C. Kitano, The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models, *Bioinformatics*, 19 (2003), p. 524-531.

[18] N. G. van Kampen, *Stochastic Processes in Physics and Chemistry*, North-Holland, Amsterdam, 1992.

[19] T. R. Kiehl, R. M. Mattheyses, M. K. Simmons, Hybrid simulation of cellular behaviour, *Bioinformatics*, 20 (2004), p. 316–322.

[20] A. Levchenko, J. Bruck, P. W. Sternberg, Scaffold proteins may biphasically affect the levels of mitogen-activated protein kinase signaling and reduce its threshold properties, *Proc. Natl. Acad. Sci. USA*, 97 (2000), p. 5818–5823.

[21] H. Lodish, A. Berk, P. Matsudaira, C. A. Kaiser, M. Krieger, M. P. Scott, S. L. Zipursky, J. Darnell, *Molecular Cell Biology*, 5th ed., Freeman, New York, 2004.

[22] P. Lötstedt, L. Ferm, Dimensional reduction of the Fokker-Planck equation for stochastic chemical reactions, *Multiscale Meth. Simul.*, 5 (2006), p. 593–614.

[23] P. Lötstedt, S. Söderberg, A. Ramage, L. Hemmingsson-Fränden, Implicit solution of hyperbolic equations with space-time adaptivity, *BIT*, 42 (2002), p. 134–158.

[24] H. H. McAdams, A. Arkin, Stochastic mechanisms in gene expression, *Proc. Natl. Acad. Sci. USA*, 94 (1997), p. 814–819.

[25] B. Moskowitz, R. E. Caflisch, Smoothness and dimension reduction in Quasi-Monte Carlo Methods, *Math. Comput. Modelling*, 23 (1996), p. 37–54.

[26] A. B. Owen, Monte Carlo variance of scrambled net quadrature, *SIAM J. Numer. Anal.*, 34 (1997), p. 1884–1910.

[27] J. Paulsson, O. G. Berg, M. Ehrenberg, Stochastic focusing: Fluctuation-enhanced sensitivity of intracellular regulation, *Proc. Natl. Acad. Sci. USA*, 97 (2000), p. 7148–7153.

[28] J. Puchalka, A. M. Kierzek, Bridging the gap between stochastic and deterministic regimes in the kinetic simulations of the biochemical reaction networks, *Biophys. J.*, 86 (2004), p. 1357–1372.

[29] C. V. Rao, A. P. Arkin, Stochastic chemical kinetics and the quasi-steady-state assumption: Application to the Gillespie algorithm, *J. Chem. Phys.*, 118 (2003), p. 4999–5010.

[30] C. V. Rao, D. M. Wolf, A. P. Arkin, Control, exploitation and tolerance of intracellular noise, *Nature*, 420 (2002) p. 231–237.

[31] L. Santoso, H. S. Booth, C. J. Burden, M. Hegland, A stochastic model of gene switches, *ANZIAM J.*, 46 (2005), p. C530–C543.

[32] P. Sjöberg, P. Lötstedt, J. Elf, Fokker-Planck approximation of the master equation in molecular biology, Technical report 2005-044, Dept of Information Technology, Uppsala University, Uppsala, Sweden, 2005, to appear in *Comput. Vis. Sci.*, available at http://www.it.uu.se/research/reports/2005-044/.

[33] M. Thattai, A. van Oudenaarden, Intrinsic noise in gene regulatory networks, *Proc. Nat. Acad. Sci. USA*, 98 (2001), p. 8614–8619.

[34] J. M. G. Vilar, H. Y. Kueh, N. Barkai, S. Leibler, Mechanisms of noise-resistance in genetic oscillators, *Proc. Nat. Acad. Sci. USA*, 99 (2002), p. 5988–5992.