Efficient computation of transient solutions of the chemical master equation based on uniformization and Quasi–Monte Carlo

ANDREAS HELLANDER[1] *

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

Abstract

A Quasi–Monte Carlo method for the simulation of discrete time Markov chains is applied to the simulation of biochemical reaction networks. The continuous process is formulated as a discrete chain subordinate to a Poisson process using the method of uniformization. It is shown that a substantial reduction of the number of trajectories that is required for an accurate estimation of the probability density functions (PDF) can be achieved with this technique. The method is applied to the simulation of two model problems. Although the technique employed here does not address the typical stiffness of such systems, it is useful when computing the PDF by replication and the method can be used in conjuncture with hybrid methods that reduce the stiffness.

Keywords: Gillespie algorithm, Quasi-Monte Carlo method, variance reduction, uniformization, stochastic chemical kinetics.

1 Introduction

Intrinsic noise can greatly affect the behavior of cellular reaction networks [33]. As a consequence, numerical simulations of such systems based on the deterministic reaction rate equations can be of limited usefulness due to the poor modeling accuracy. The chemical system is often modeled as a continuous–time discrete–space Markov (CTMC) process. Thus, a better description of the system behavior is obtained from the Master equation, a differential-difference equation that describes the time evolution of the probability density. Unfortunately, direct numerical solution of this equation suffers from an unfavorable exponential

*Financial support has been obtained from the Swedish Graduate School of Mathematics and Computing.

growth in computational time with the number of species in the model. For low dimensional problems, fast direct solution methods have been developed [29, 7, 14, 26, 36], but for larger systems the only viable alternative is simulation with the Stochastic Simulation Algorithm (SSA) [10]. It yields correct realizations of the underlying stochastic process, and by replication quantities such as expected values and covariances can easily be estimated. Being a Monte Carlo method, SSA does not suffer from the curse of dimensionality. However, the convergence rate is of order $M^{-1/2}$ where M is the number of replications. For systems with large variance in the copy numbers, many trajectories must be generated to compute a good approximation of the probability density function (PDF). As a consequence, models with species that differ greatly in numbers and/or display large differences in rate constants may be very difficult to simulate. Also, the performance is very sensitive to the stiffness in the systems arising from large differences in time scales of the reactions. This problem has been recognized by the community, and much effort has been put into methods aimed at finding a remedy to the issue of stiffness. Quite a few hybrid and multiscale methods have emerged, that make use of different simplifying assumptions in order to speed up simulations [28, 13, 6, 3, 2, 11, 15]. In many cases, they provide good alternatives when the system does not permit simulation with the classical SSA.

Sometimes it is sufficient for the modeler to study the behavior of single trajectories to draw conclusions concerning the behavior of the system at hand. For those cases the above mentioned methods or even the unmodified SSA will be sufficient. However, it is equally easy to imagine situations where the complete picture available from the entire PDF would be useful. Whenever we are interested in the time dependent behavior of the system and want to compute for example the PDF, we must resort to replication, and as a consequence of the poor convergence rate of Monte Carlo methods, a large number of trajectories have to be sampled. Even if the system is not particularly stiff, or if the stiffness has been taken care of by a hybrid method, this can still be very costly, resulting in prohibitively long simulation times. Little attention has been given to this issue, although a recent exception can be found in [24].

In related fields e.g. computational finance the use of variance reduction techniques for the simulation of Markov Chains have been investigated. One such technique with the potential of improving the convergence rate of MC simulations is the use of Quasi-Monte Carlo (QMC) methods. Recently, methods that use quasi-random sequences in order to simulate discrete–time Markov chains (DTMCs) have been developed [19, 21, 12, 20]. Examples in this paper show that simulation times are reduced with very simple means by the adaption of the QMC methods to SSA. The main problem encountered is that we need to deal with CTMCs. The QMC method is adapted to the simulation of biochemical reaction networks by using the well known technique of uniformization for CTMCs. This construction has been used in other fields, e.g. to develop efficient parallel

simulation methods and together with variance reduction methods [31, 32, 35, 27] and recently in the context of solving the master equation with a direct method [36]. The purpose of this work is to illustrate that the combination of uniformization and Quasi–Monte Carlo methods can speed up simulations with SSA. This will be done by applying the algorithm to two numerical examples. In those cases we will see that the uniformized QMC method is almost 30 times faster than SSA, when the method is applied to the problem of computing transient solutions of the PDF through replication.

2 The Stochastic Simulation Algorithm

In this section a brief description of SSA [10] will be given. We consider a chemical system consisting of S species and R reactions. We let $\mathbf{x}$ be the S dimensional state vector. Chemical reactions occurs with propensities $\mu_r(\mathbf{x})$. Simulation of one step of trajectory j amounts to sampling an inter event time τ_n from an exponential distribution with parameter $\lambda_j = \sum_{r=1}^{R} \mu_r(\mathbf{x}_j)$ and choosing the reaction that occur after that time with probability $\mu_r(\mathbf{x}_j)/\lambda_j$. In order to do this, we need to generate two uniformly distributed random numbers $u_1, u_2 \in U(0,1)$. We want to estimate the PDF at a deterministic stopping time T, that we choose. The simulation of one trajectory is conducted as

1. Compute $\lambda_j = \sum_{r=1}^{R} \mu_r(\mathbf{x}_j)$.
2. Generate the inter event time $\tau_n = -\log(u_1)/\lambda_j$.
3. Find r s.t. $\sum_{i=1}^{r-1} \mu_i < \lambda_j u_2 < \sum_{i=1}^{r} \mu_i$.
4. Execute reaction r, i.e. update the state vector according to the transition rule corresponding to r.
5. $t = t + \tau_n$.
6. Repeat while $t < T$.

The algorithm above corresponds to the direct method [10]. Several improvements of the original implementation of the algorithm have been proposed, differing mainly in the ways the next transition is chosen and in the way the state is updated. Among them is the method due to Gibson and Bruck [9], the optimized and the sorting direct method (ODM,SDM) [4, 25] and the logarithmic direct method (LDM) [23].

3 Uniformization

Here we will describe how uniformization [17] can be used to simulate the systems with SSA. We will need this construction in order to apply the QMC algorithm [21] to SSA, as explained in Sect. 5. Basically, by uniformization we make the M replicated chains have i.i.d inter–event times independent of the particular state of the individual chains, i.e. τ_n is exponentially distributed, $\tau_n \in Exp(\lambda_{max})$. λ_{max} is a parameter we choose, as will be explained below. In this way we obtain a DTMC subordinate to a Poisson process with inter–event times distributed as above. This can be achieved by adding a null transition to the chain. More formally, let $\{X_t, t \geq 0\}$ be our CTMC. We assume that the infinitesimal generator $\mathbf{Q}$ is bounded, i.e. that $\sup_i -Q_{ii} < \infty$. Denote an upper bound on $-Q_{ii}$ in the interval $[0, T]$ by λ_{max}. If $\{Z_n\}$, $n \in \mathbb{Z}_+$ is the DTMC with transition probability matrix $\mathbf{D} = \mathbf{I} + \frac{1}{\lambda_{max}}\mathbf{Q}$, then the process $\{X_t, t \geq 0\}$ and the process $\{Z_{N_t}, t \geq 0\}$ have the same distributions. A simple proof of uniformization can be found in [5].

The propensity functions $\mu_i(\mathbf{x})$ will typically be bounded functions of $\mathbf{x}$, and thus the process can be uniformized. Simply choose λ_{max} such that $\lambda_j \leq \lambda_{max}, 1 \leq j \leq M, \forall t \in [0, T]$. In practice this can be done with presimulation of a few trajectories. Introduce a null transition $\mathbf{x} \to \mathbf{x}$, and set the reaction propensities as $\tilde{\mu}_i(\mathbf{x}_j) = \mu_i(\mathbf{x}_j)/\lambda_{max}, \quad i = 1, \ldots, R$, and the null transition $\tilde{\mu}_{R+1}(\mathbf{x}_j) = 1 - \sum_{i=1}^{R} \tilde{\mu}_i(\mathbf{x}_j)$.

4 Estimating the PDF

Given a system with state space Ω of N states (generally Ω can be the whole positive quadrant, but in practice, for finite times, it will be a finite subdomain) and M trajectories generated with SSA as in Sect. 2, the PDF at a point $\mathbf{x}_i \in \Omega$, $p(\mathbf{x}_i, t)$, is estimated by

$$p(\mathbf{x}_i, t) \approx \hat{p}(\mathbf{x}_i, t) = \frac{1}{M} \sum_{j=1}^{M} \chi(\mathbf{x}_i, \mathbf{x}_j) \tag{1}$$

where $\chi(\mathbf{x}_i, \mathbf{x}_j)$ is the characteristic function

$$\chi(\mathbf{x}_i, \mathbf{x}_j) = \begin{cases} 1 & \text{if} \quad \mathbf{x}_i = \mathbf{x}_j \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

Given J replications the pooled mean and the empirical variance is given by

$$\bar{p}(\mathbf{x}_i, t) = \frac{1}{J} \sum_{j=1}^{J} \hat{p}_j(\mathbf{x}_i, t), \quad i = 1 \ldots N, \tag{3}$$

$$s_i^2 = \frac{1}{J-1}\sum_{j=1}^{J}(\hat{p}_j(\mathbf{x}_i,t) - \bar{p}(\mathbf{x}_i,t))^2, \quad i = 1\ldots N. \tag{4}$$

The situation is somewhat different if we instead simulate the uniformized chain. We first note that the number of events N_T taking place in the interval $[0,T]$ is Poisson distributed according to $N_T \in Po(\lambda_{max}T)$. Thus the PDF at time T can be written

$$p(\mathbf{x}_i,T) = \sum_{k=0}^{\infty} Pr(N_T = k)p_k(\mathbf{x}_i,k). \tag{5}$$

where $p_k(\mathbf{x}_i,k)$ is the distribution at step k of the DTMC $\{Z_n, n \geq 0\}$. This distribution is approximated by SSA for the modified system in the same manner as in (1). In practical calculations the sum in (5) must be truncated. We choose u_l, u_r such that $1 - \sum_{k=u_l}^{u_r} Pr(N_T = k) < \varepsilon$. This can be done efficiently and in a numerically stable manner with e.g. the Fox-Glynn algorithm [8]. Then the approximation of the PDF will be

$$p(\mathbf{x}_i,T) \approx \hat{p}(\mathbf{x}_i,t) = \sum_{k=u_l}^{u_r} Pr(N_T = k)\hat{p}_k(\mathbf{x}_i,k) = \frac{1}{M}\sum_{k=u_l}^{u_r} Pr(N_T = k)\sum_{j=1}^{M}\chi(\mathbf{x}_i,\mathbf{x}_j). \tag{6}$$

The error in this approximation can obviously be estimated as in (4). The l_2 norm of $\mathbf{s}^2$ with components as in (4) will be the quantity used for comparing the efficiency of the methods.

The obvious drawback with the uniformized simulation is that the number of iterations will be larger than in the direct method since we uniformize with the fastest scales. This means that the approach will be less suitable for stiff systems. Also, in order to estimate the PDF at time T, the PDFs of the discrete chain will have to be evaluated $u_r - u_l$ times. This number increases with both λ_{max} and T. On the other hand, the simulation of the process $\{Z_n, n \geq 0\}$ only requires the generation of one random number per step. In addition, we will be able to apply the QMC method in Sect. 5 to the estimation of $p_k(\mathbf{x}_i,k)$ in (5), providing variance reduction and in some cases an improved convergence rate. Also, simulation based on uniformization give better resolution in the approximation of the PDF even when applied without any variance reduction method. In some cases this effect is already enough to motivate the use of the uniformization procedure.

5 Quasi-Monte Carlo method

In this section we will describe how the QMC algorithm in [21] can be efficiently applied to simulation with SSA. In contrast to MC methods, QMC methods make use of low-discrepancy point sets in order to improve the convergence rate and reduce the variance of estimators. They have been successfully applied to numerical integration in high dimensions and the solution of stochastic differential equations and are widely used in e.g. the field of computational finance. For an overview see [22]. However, since they unlike MC methods are constructed so that the points in a sequence are correlated, they are harder to apply to the simulation of stochastic processes. Simply replacing pseudo-random sequences by QMC sequences in the algorithm will not work for this reason.

Simulation strategies using low-discrepancy point sets have been developed for the simulation DTMCs [19, 21, 12]. For a review see [20]. Most of the methods use a sequence of dimensionality dN_T to simulate the discrete time chain N_T steps if the transitions require d uniform random numbers. QMC methods tend to be less efficient in very high dimensions. In addition, the generation of high dimensional sequences is expensive. Typically the simulation of a biochemical system will require much higher dimensionality of the sequences than most sequence generators can provide. As a consequence, these methods will currently be of limited use in this context. However, the algorithm given in [21] uses a single randomized quasi random sequence and can easily be applied to simulation with a large number of steps. This method will be explored here. One may think of applying the QMC algorithm to SSA as in Sect. 2 directly when simulating the embedded chain. This will not work though, since every trajectory will need a different number of time steps to reach the given stopping time. Correlation between individual chains in the same step is introduced by the QMC method, but not between trajectories at diffent times. The problem arises when trajectories start reaching the stopping time T. As trajectories finish, shorter and shorter sequences will be used and this will eventually destroy the variance reduction which is noted in [21]. For this reason we use uniformization as described in Sect. 3 to estimate the PDF at time T as in (5). Since we want to compute the solution at time T, the QMC method does not need to be applied in all k steps in the simulation of the uniformized chain, if k is sufficiently large. We will evaluate distributions in the interval $[u_l, u_r]$, so we use quasi-sequences in the interval $[u_l - N_{QMC}, u_r]$, where we choose N_{QMC}. In this setting the application of the algorithm given in [21] to SSA will be as follows

1. (Initialize) Select a uniformization rate valid in the interval $[0, T]$. Initialize a d-dimensional quasi-random sequence $\mathbf{S_Q}$ and a pseudo random sequence $\mathbf{S_P}$ of length M. Compute the Poisson probabilities with error tolerance ε to yield u_l and u_r. Choose N_{QMC}. Initialize the chains.

2. for $k = 0 : u_r$

(a) if $k < N_{QMC}$

i. Simulate the M chains one step with SSA (uniformized) using $\mathbf{S_P}$.

(b) else

i. Simulate the M chains one step with SSA (uniformized) using $\mathbf{S_Q}$.

ii. Reorder the trajectories in increasing order of the states.

iii. Scramble the sequence $\mathbf{S_Q}$.

iv. if $k \geq u_l$

A. Compute $\hat{p}_k(\mathbf{x}, k)$ using the M trajectories.

B. Add $\hat{p}_k(\mathbf{x}, k)$ to the sum in (5).

3. Repeat the procedure J times to compute a pooled mean and an empirical variance of the estimator as in (4).

It is shown in [21] that the method yields unbiased estimators of averages and empirical variances. There, only models with a one-dimensional state space are examined. We will typically consider models in higher dimensions. This is the topic of [12], but that method extends [19] and it does not use randomized sequences. Step (3) in the algorithm above requires the trajectories to be sorted in increasing order. One can think of several ways to define the ordering, but here a component–wise sorting is used, i.e. first the trajectories are sorted according to the first component, then according to the second etc. The order in which the molecules are sorted can obviously be chosen arbitrarily, though some experiments made suggest that the ordering affects the results. Also, it may not be necessary to sort every coordinate if the dimensionality d is high. This is not pursued in this paper but can be subject for further detailed studies of the method. The trajectories are stored in a $(M \times d)$ matrix which is sorted using a least significant digit (LSD) radix sort where each column is sorted using ProxMap sort [30]. This gives an average runtime $\mathcal{O}(M)$. The keys (rows in the matrix) are of fixed, equal length and the complexity of the sorting step is $\mathcal{O}(Md)$. Typically, in every iteration species can change in copy numbers one or two steps, so changes in the sorted trajectories will tend to be local. It is possible to use information in the stoichiometry matrix of the system to make the sorting step even more efficient. This, however, has not been done here.

A consequence of the uniformization is that only one random number is needed for the SSA step (2). Also, in a sense we use more information when we estimate the PDF weighting with the Poisson probabilities. This means that the resolution of the PDF will be better than that obtained from the direct method even without the QMC extension (at the price of evaluating the PDF more times). This will be illustrated in Sect. 6. There, the direct method and the uniformized QMC algorithm above will be compared for two model problems. The empirical variances of 2-D marginal distributions will be computed, but since the state space is sorted, distributions of higher dimensionality can be evaluated in selected points in the state space efficiently as in [15], with low additional cost.

6 Numerical experiments

In this section the method will be applied to two model problems. The first one is a system in four dimensions, where the large variation in the copy number of certain species require the sampling of a large number of trajectories in order to arrive at the desired accuracy. In Sect. 6.2 an example of a MAPK kinase cascade in 8 dimensions is considered. Also for this example the uniformized QMC method give good results. Both examples are non-stiff in the sense discussed in Sect. 1 due to the limitations discussed in sec. 3. However, in Sect. 7 we argue that the method is useful in a more general setting.

In all experiments, the randomized QMC sequences are generated by the algorithm described in [16]. Throughout, Sobol sequences with modified Owen's scrambling are employed. Poisson probabilities are computed with the Matlab routine `poisspdf` but could in a later implementation be computed as in [8]. The majority of the code was implemented in C and wrapped as mex-files to be called from Matlab. All numerical experiments were conducted on a MacBook Core Duo 2.0 Ghz with 2 MB L2 cache and 2 Gb of RAM.

6.1 Coupled flows

Here the uniformized QMC method is applied to a simple model with two metabolites whose production is controlled by a corresponding enzyme. The reactions of the model are

$$\begin{aligned}
&\emptyset \xrightarrow{\frac{k_a e_A}{1+\frac{a}{K_i}}} A \qquad && \emptyset \xrightarrow{\frac{k_b e_B}{1+\frac{b}{K_i}}} B \\
&A + B \xrightarrow{k_2 ab} \emptyset \\
&A \xrightarrow{\mu a} \emptyset && B \xrightarrow{\mu b} \emptyset \\
&\emptyset \xrightarrow{\frac{k_e e_A}{1+\frac{a}{K_r}}} E_A && \emptyset \xrightarrow{\frac{k_e e_B}{1+\frac{b}{K_r}}} E_B \\
&E_A \xrightarrow{\mu e_A} \emptyset && E_B \xrightarrow{\mu e_B} \emptyset
\end{aligned}$$

where the reaction constants take the values in Table. 1.

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.

This system is not stiff, but a good resolution of the PDF requires a large number of trajectories due to large fluctuations of the metabolite species A and B. The relative fluctuations in the enzyme species is small compared to those of

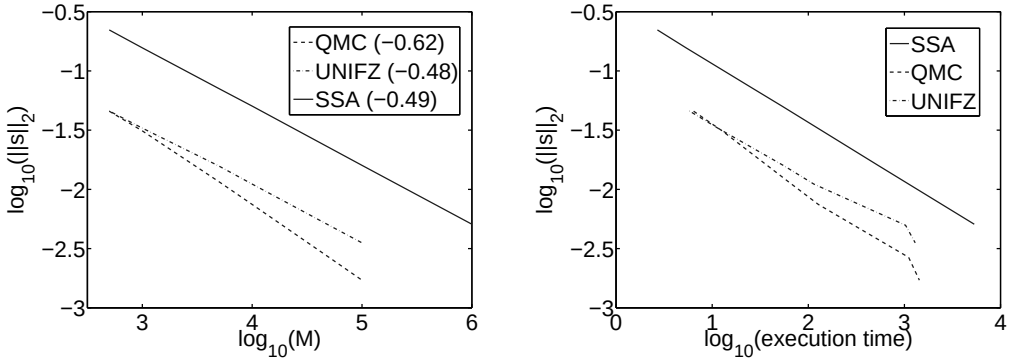

Fig. 1: A. Hellander., Fig. 1

the metabolites. The error in the marginal PDF of A and B is measured. The species is sorted in the order A, B, E_a, E_b.

As discussed in Sect. 2, there are many ways to implement SSA. In order to make a fair comparison, we must compare the QMC method to the best performing implementation. We will deal with fairly small systems, aiming at the probability distribution, and thus the next reaction method (NRM) of Gibson and Bruck [9] will most likely be slower than the direct method (DM) due to the overhead of maintaining the indexed priority queue and the dependency graph. The performance of DM depends on the ordering of the reactions, and ODM [4] will always perform better. It is the method used here. Also, the order of the most frequently occurring reactions does not change drastically during the course of simulation, so SDM [25] is not expected to perform better than ODM and was not implemented.

The system was simulated with varying number of trajectories up to time $t = 100s$. The uniformization rate was $\lambda_{max} = 10$ and $u_l = 850, u_r = 1150$, yielding $\varepsilon = 2.17 \cdot 10^{-6}$. N_{QMC} was chosen to be 800.

To the left in Fig. 1 we show the convergence rate of the norm of $\mathbf{s}$ as computed in (4). As can be seen, the convergence rate of the QMC method is higher than the pseudo–random SSA, although not as high as expected for general QMC methods. To the right in the same figure we plot the norm of $\mathbf{s}$ against the simulation time for SSA and QMC.

M (30×10^3)	1	10	100
UNIFZ	24	21	20
QMC	29	53	101

Table 2: Variance reduction factors α_{UNIFZ} and α_{QMC}. The effect of QMC is clearly visible for the higher number of trajectories.

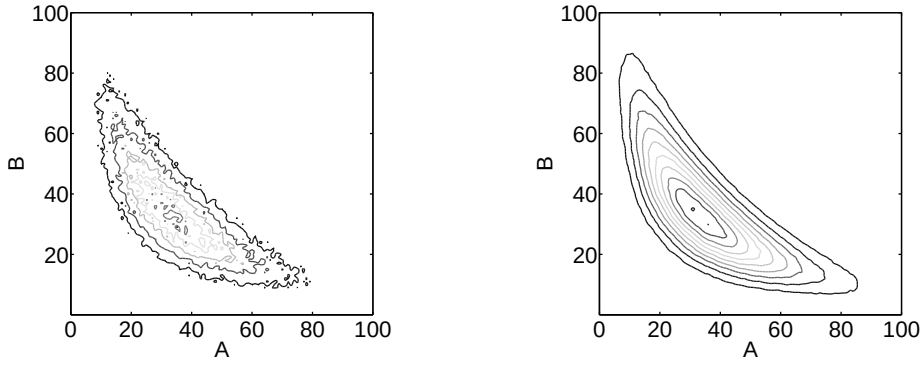

Fig. 2: A. Hellander., Fig. 2

Table 6.1 shows the variance reduction factor for increasing M, i.e. $\alpha_{QMC} = \|\mathbf{s}^2_{SSA}\|_2/\|\mathbf{s}^2_{QMC}\|_2$ and $\alpha_{UNIFZ} = \|\mathbf{s}^2_{SSA}\|_2/\|\mathbf{s}^2_{UNIFZ}\|_2$. As can be seen, for 10^5 trajectories α is approximately 100. This means that in order to achieve the same error with SSA, we have to increase the number of trajectories by a factor 100. The time to simulate 10^5 trajectories with the QMC method and with SSA was 48s and 18s. resp. The speedup is then $\alpha_{QMC}t_{SSA}/t_{QMC} \approx 33$ if we want to achieve the same accuracy with SSA.

At simulation time $t = 100s$ the state space of A, B is relatively small. For larger state spaces, with more variation in the copy numbers, the effect of QMC is small for small values of M. It starts to show when we increase the number of trajectories, i.e. use longer quasi-sequences. This can be understood since we need to approximate the PDF at every point using a fair number of trajectories if the variance reduction is to have effect. An example of this scenario is if we simulate the same system to the time $t = 1000s$, where the marginal PDF of A and B has developed a more stretched shape. Here we use $M = 30 \times 10^5$ trajectories to clearly see the effect of QMC. In this case the variance is reduced a factor 2.5 compared to the uniformized simulation giving a variance reduction factor of 79 compared to SSA. The speedup factor is approximately 28. In practice, when the system is simulated to time $t = 1000s$, with this implementation the QMC calculation takes 3.14h, while the SSA calculation would take about 79h (estimated). Even though these calculations are not very demanding, the difference is considerable in daily work. Fig. 2 shows the marginal distribution computed with SSA and the QMC method using the same number of trajectories, and clearly the solution computed with the latter method is better.

6.2 A MAPK cascade with feedback

Here an example in 8 dimensions is simulated, modeling a MAPK cascade with feedback [18]. With parameters in a certain range, the model produces sustained oscillations in the activated and inactivated form of MAPK.

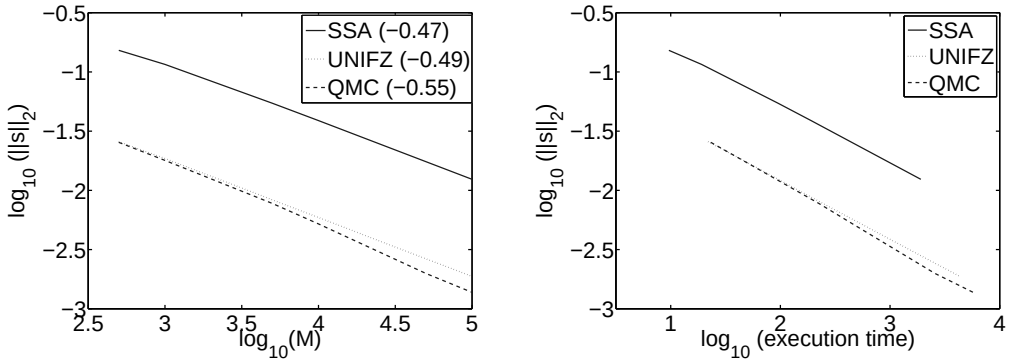

Fig. 3: A. Hellander., Fig. 3

The system was simulated with the same parameter values used to produce Fig. 2A in [18] with ODM and the uniformized QMC method. Since the system exhibits periodic oscillations, the maximal value of λ_j will also oscillate. But the propensities do not take very large values, and a value $\lambda_{max} = 15$ is sufficient. This was determined by presimulation with few trajectories. The marginal PDF of the species MAPK and MAPKPP (doubly phosphorylated) is computed at time $t = 200s$.

Table 6.2 and Fig. 3 shows results from this simulation. We clearly see that also for this system the variance reduction is substantial, at least for large values of M. Here we used $u_r = 3500$, $u_l = 2500$ and $N_{QMC} = 2480$, giving $\varepsilon = 2.5 \cdot 10^{-12}$. Also here, a slightly improved convergence rate is obtained, but smaller than for the example in Sect. 6.1. Even so, the execution time is improved with the uniformized QMC method.

For the largest value of M in Tab. 6.2, $\alpha_{QMC} \approx 2\alpha_{UNIFZ}$, but to the right in Fig. 3 we see that the difference in performance of the methods is small. In this case it may be sufficient to use uniformization only, but QMC never performs worse than the uniformized simulation. If we would simulate the system with SSA to an error of the magnitude obtained with the QMC method ($M = 30 \times 10^5$), the QMC method is 29 times faster while the corresponding speedup with uniformization only is 20.

M (30×10^3)	1	10	100
UNIFZ	46	45	44
QMC	49	60	86

Table 3: Variance reduction factors α_{UNIFZ} and α_{QMC}.

Fig. 4 shows the marginal PDF at time $t = 200s$ computed with 10^5 trajectories. Also for this example, it is evident that the QMC method produces a better solution.

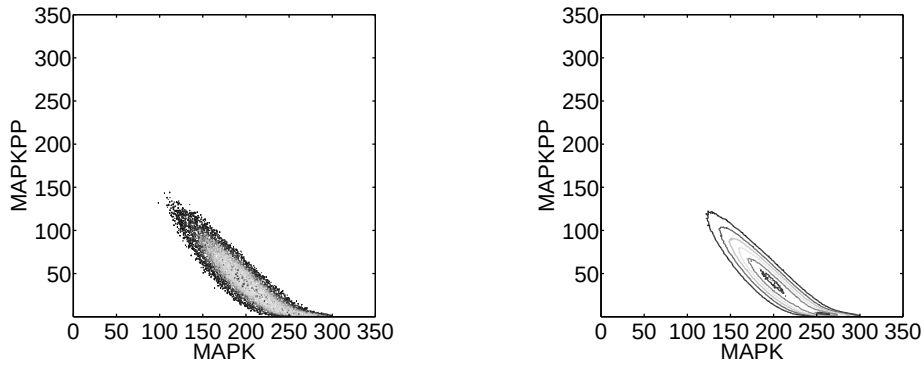

Fig. 4: A. Hellander., Fig. 4

7 Conclusions and discussion

It is shown that uniformization, combined with a QMC method [21], can be efficiently applied to the simulation of biochemical reaction networks when the aim is to compute transient solutions of the PDF. Two numerical examples have been considered, and the QMC method yields superior results.

The code developed in this paper is serial. Since SSA is easily parallelized it is important that the uniformized QMC method can be efficiently implemented in parallel. The uniformized chain is even better suited for simulation in parallel. Parallel generation of quasi random numbers are considered in[1]. One direction of future work is to make a parallel implementation of the method to ensure that it is still superior to SSA in that setting.

The issue of stiffness is addressed in e.g. [34, 35] where adaptive uniformization is investigated. It would certainly be interesting and possible to incorporate that method in this context. This could make the range of systems amenable to QMC simulation larger. Foremost, the uniformized QMC method could be useful in conjuncture with other methods that reduce the stiffness, e.g. MSSA [2], nested SSA [6] and other. The hybrid method proposed by the author in [15] uses dimension reduction to arrive at a relatively small set of stochastic variables of which the PDF is computed during the course of simulation. For such reduced systems that will be non–stiff and with a smaller state space than the original model, the QMC method is expected to be useful.

8 Acknowledgments

Many discussions with Per Lötstedt have improved the content of this paper. I also thank Stefan Engblom for valuable comments. Financial support has been provided by the Swedish Graduate School in Mathematics and Computing.

References

[1] B.C. Bromley. Quasirandom number generators for parallel Monte Carlo algorithms. *J. Parallel and Distributed Computing*, 38(1):101–104, 1995.

[2] Yang Cao, Dan Gillespie, and Linda Petzold. Multiscale stochastic simulation algorithm with partial equilibrium assumption for chemically reacting systems. *J. Comput. Phys.*, 206:395–411, 2005.

[3] Yang Cao, Daniel T Gillespie, and Linda R Petzold. The slow-scale stochastic simulation algorithm. *J. Chem. Phys.*, 122, 014116, 2005.

[4] Yang Cao, Hong Li, and Linda Petzold. Efficient formulation of the stochastic simulation algorithm for chemically reacting systems. *J. Chem. Phys*, 121(9):4059–4067, 2004.

[5] Nico M. Van Dijk. On a simple proof of uniformization for continuous and discrete-state continuous-time Markov chains. *Adv. Appl. Probab.*, 22(3):749–750, 1990.

[6] Weinan E, Di Liu, and Eric Vanden-Eijnden. Nested stochastic simulation algorithm for chemical kinetic systems with disparate rates. *J. Chem. Phys.*, 123, 194107, 2005.

[7] Stefan Engblom. A discrete spectral method for the chemical master equation. *Technical Report 2006-036, Dept. Information Technology, Uppsala University, Uppsala, Sweden, avaliable at: http://www.it.uu.se/research/publications/reports/2006-036/*, 2006.

[8] Bennet L. Fox and Peter W.Glynn. Computing Poisson probabilities. *Communications of the ACM*, 31(4):440–445, 1988.

[9] Michael A. Gibson and Jehoshua Bruck. Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem.*, 104:1876–1889, 2000.

[10] Daniel T Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reacting systems. *J. Comput. Phys.*, 22:403–434, 1976.

[11] Daniel T Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115(4):1716–1733, 2001.

[12] Rami El Haddad, Christian Lécot, and Pierre L'Ecuyer. Quasi-Monte Carlo simulation of discrete-time Markov chains on multidimensional state spaces. *available at http://www.iro.umontreal.ca/ lecuyer/myftp/papers/multiqmc06.pdf*, 2006.

[13] Eric L Haseltine and James B Rawlings. Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *J. Chem. Phys*, 117(15):6959–6969, 2002.

[14] M Hegland, C Burden, L Santoso, S MacNamara, and H Booth. A solver for the stochastic master equation applied to gene regulatory networks. *J. Comput. Appl. Math*, 205:708–724, 2005.

[15] Andreas Hellander and Per Lötstedt. Hybrid method for the chemical master equation. *J. Comput. Phys.*, doi: 10.1016/j.jc.2007.07.020 ., 2007.

[16] Hee Sun Hong and Fred J. Hickernell. Algorithm 823: Implementing scrambled digital sequences. *ACM Trans. Math. Soft*, 29(2):95–109, 2003.

[17] A Jensen. Markoff chains as an aid in the study of Markoff processes. *Skand. Aktuarietidskr.*, 36:87–91, 1953.

[18] Boris N. Kholodenko. Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. *Eur. J Biochem.*, 267:1583–1588, 2000.

[19] Christian Lécot and Bruno Tuffin. Quasi–Monte Carlo methods for estimating transient measures of discrete time Markov chains. In Harald Neiderriter, editor, *Monte Carlo and Quasi Monte Carlo Methods*. Springer-Verlag, 2002.

[20] Christian Lécot and Bruno Tuffin. Comparison of Quasi-Monte Carlo-based methods for the simulation of Markov chains. *Monte Carlo Methods and Applications*, 10:337–384, 2004.

[21] Pierre L'Ecuyer, Christian Lécot, and Bruno Tuffin. A randomized Quasi-Monte Carlo simulation method for Markov chains. *in Monte Carlo and Quasi-Monte Carlo Methods 2004*, pages 331–342, 2007.

[22] Christiane Lemiuex and Pierre L'Ecuyer. On the use of Quasi-Monte Carlo methods in computational finance. In *Lecture notes in Computer Science*, number 2073. Springer, New York, 2001.

[23] Hong Li and Linda Petzold. Logarithmic direct method for discrete stochastic simulation of chemically reacting systems. *availabe at: http://www.engineering.ucsb.edu/ cse/Files/ldm0513.pdf*, 2006.

[24] Azi Lipshtat. An "all possible steps" approach to the accelerated use of Gillespie's algorithm. *J. Chem. Phys.*, 126, 184103, 2007.

[25] James M McCollum, Gregory D Peterson, Chris D Cox, Michael L Simpson, and Nagiza F Samtova. The sorting direct method for stochastic simulation of biochemical systems with variying reaction execution behaviour. *Computational Biology and Chemistry*, 30:39–49, 2005.

[26] Brian Munsky and Mustafa Khammash. The finite state projection algorithm for the solution of the chemical master equation. *J. Chem. Phys.*, 124(044104), 2006.

[27] David M. Nicol and Philip Heidelberger. Parallel algorithms for simulating continuous time Markov chains. *ACM SIGSIM Simulation Digest*, 23:11–18, 1993.

[28] Howard Salis and Yiannis Kaznessis. Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions. *J. Chem. Phys.*, 122, 054103, 2005.

[29] Paul Sjöberg, Per Lötstedt, and Johan Elf. Fokker-Planck approximation of the master equation in molecular biology,. *Comput. Vis. Sci. (to appear)*, 2007.

[30] Thomas A. Standish. *Data Structures in Java*. Addison-Wesley, 1998.

[31] Simon Streltsov and Pirooz Vakili. Parallel replicated simulation of Markov chains: Implementation and variance reduction. *Proceeings of the 1992 Winter Simulation Conference*, pages 430–436, 1993.

[32] Simon Streltsov and Pirooz Vakili. Variance reduction algorithm for parallel replicated simulation of uniformized Markov chains. *Discrete Event Dynamic Systems: Theory and Applications*, 6:159–180, 1996.

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

[34] Aad P.A. van Moorsel and William H. Sanders. Adaptive uniformization. *Commn. Statist.-Stochastic Models*, 10(3):619–647, 1994.

[35] Aad P.A. van Moorsel and William H. Sanders. Transient solution of Markov models by combining adaptive & standard uniformization. *IEEE Transactions on Reliability*, 46(3):430–440, 1997.

[36] Jingewei Zhang, Layne T Watson, and Yang Cao. A modified uniformization method for the solution of the chemical master equation. *avaliable at: http://eprints.cs.vt.edu/archive/00000992/01/IETSB07.pdf*, 2007.