Summary of papers for QMC applied to the τ-leap algorithm

Manuel Holschbach

January 23, 2025

Recap: Stochastic reaction networks

- We aim to model stochastic reaction networks (SRN), consisting of n different chemical species $S_1, \ldots, S_n$ that interact via K different reactions $R_1, \ldots, R_K$, that can be described as

$$\alpha_{1,k} S_1 + \ldots + \alpha_{n,k} S_n \xrightarrow{c_k} \beta_{1,k} S_1 + \ldots + \beta_{n,k} S_n \quad k = 1, \ldots, K$$

  where c_k is the reaction rate constant for the k-th reaction
- One way to model a SRN is through a continuous time Markov Chain, which leads to the Kurtz random time change representation

$$X(t) = X(0) + \sum_{k=1}^{K} Y_k\left(\int_0^t a_k(X(s))ds\right)\zeta_k$$

Chemical Langevin equation

- We can rewrite this equation as

$$X(t+\tau) = X(t) + \sum_{k=1}^{K} Y_k\left(\int_t^{t+\tau} a_k(X(s))ds\right)\zeta_k$$

- Assuming a small enough τ such that $a_k(X(s)) \approx a_k(X(t))$, we can approximate the Markov chain by

$$X(t+\tau) \approx X(t) + \sum_{k=1}^{K} Y_k(a_k(X(t))\tau)\zeta_k$$

Chemical Langevin equation

- If we further assume that $a_k(X(t))\tau >> 1$, we can approximate the Poisson processes by normal random variables

$$X(t+\tau) \approx X(t) + \sum_{k=1}^{K} [a_k(X(t))\tau + \sqrt{a_k(X(t))\tau}\mathcal{N}_k(0,1)]\zeta_k$$

- Taking the limit $\tau \to 0$, we obtain the chemical Langevin equation

$$dX_t = \left[\sum_{k=1}^{K} a_k(X_t)\zeta_k\right] dt + \sum_{k=1}^{K} \sqrt{a_k(X_t)}\zeta_k dW_{t,k}$$

- We can simulate this SDE using e.g. the Euler-Maruyama method

τ-leap approach

- We consider again the equation

$$X(t+\tau) \approx X(t) + \sum_{k=1}^{K} Y_k(a_k(X(t))\tau)\zeta_k$$

- Based on this equation, we obtain the update step for the τ-leap method

$$\hat{X}(t+\tau) = \hat{X}(t) + \sum_{k=1}^{K} \mathcal{P}(a_k(\hat{X}(t))\tau)\zeta_k$$

  where $\mathcal{P}(a_k(X(t))\tau)$ are Poisson random variables
- Using the resulting sample paths, we can then approximate a quantity of interest by

$$\mathbb{E}[g(X(t))] \approx \frac{1}{N} \sum_{n=1}^{N} g(\hat{X}^{(n)}(t))$$

QMC-approach

- We aim to use quasi Monte Carlo (QMC) methods to improve the convergence rate of the statistical error
- To this end, we replace the random sequence by a low-discrepancy sequence $\{v_n\}_{n=1}^{N}$ (e.g. the Sobol sequence) combined with M random shifts
- In the case of the τ-leap method, this yields the update step

$$\hat{X}_{n,m}(t+\tau) = \hat{X}_{n,m}(t) + \sum_{k=1}^{K} F^{-1}(v_{n,k}^{(m)}, a_k(\hat{X}_{n,m}(t))\tau)\zeta_k$$

- Observe that the dimensionality of the low-discrepancy points is given by $K\lceil \frac{T}{\tau} \rceil$

Numerical results

- We compare the results from the τ-leap method with the results from an Euler-Maruyama discretization of the CLE
- We consider a simple chemical reaction network given by

$$S_1 \xrightarrow{c} \emptyset$$
$$S_1 \xrightarrow{c} 2S_1$$

- Observe that in this case $\mathbb{E}[X(t)] = X(0)$ and $Var[X(t)] = 2ctX(0)$

Numerical results

- For the first 10^2 samples, the RQMC τ-leap approach has a convergence rate of $\mathcal{O}(N^{-1})$
- Afterwards, the convergence rate seems to switch to $\mathcal{O}(N^{-\frac{1}{2}})$, similar to the standard MC approach

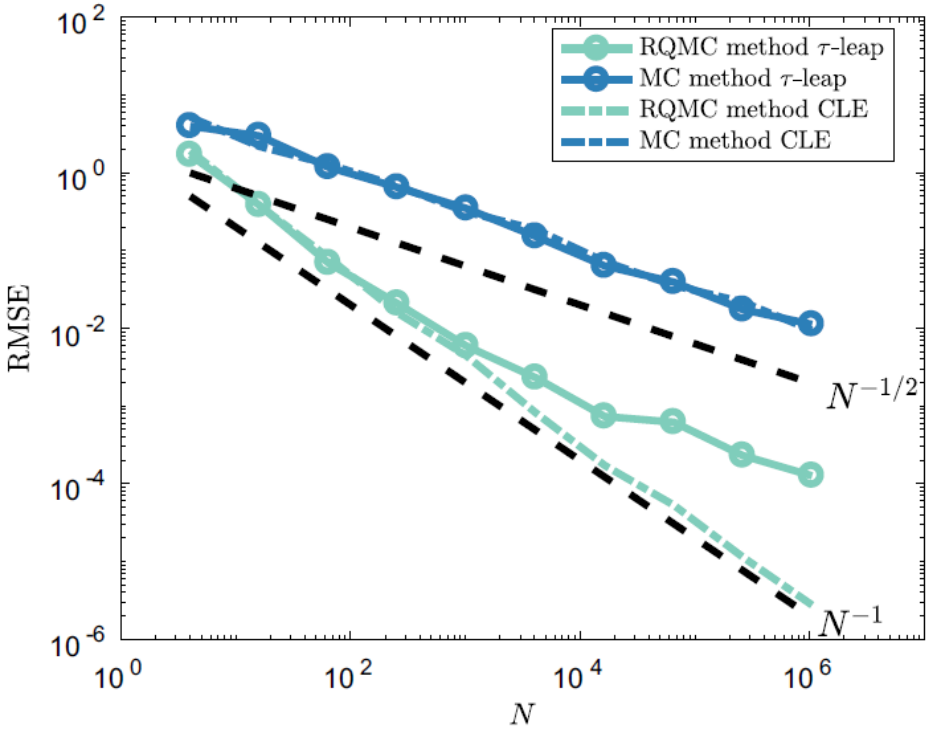

Figure: Comparison of RMSE convergence for RQMC and MC methods of τ-leap and CLE approaches for the approximation of $\mathbb{E}[S_1]$. The chosen parameters are $X(0) = 10^3$ and $c = 1$.

Numerical results

- We now consider the same reaction network for $X(0) = \varepsilon^{-1}$
- It can be seen that for lower values of ε, i.e. higher values for the number of S_1 molecules, the switch to a lower convergence rate happens after a higher number of samples
- As such, the RQMC τ-leap method performs better as the number of molecules in the system increases

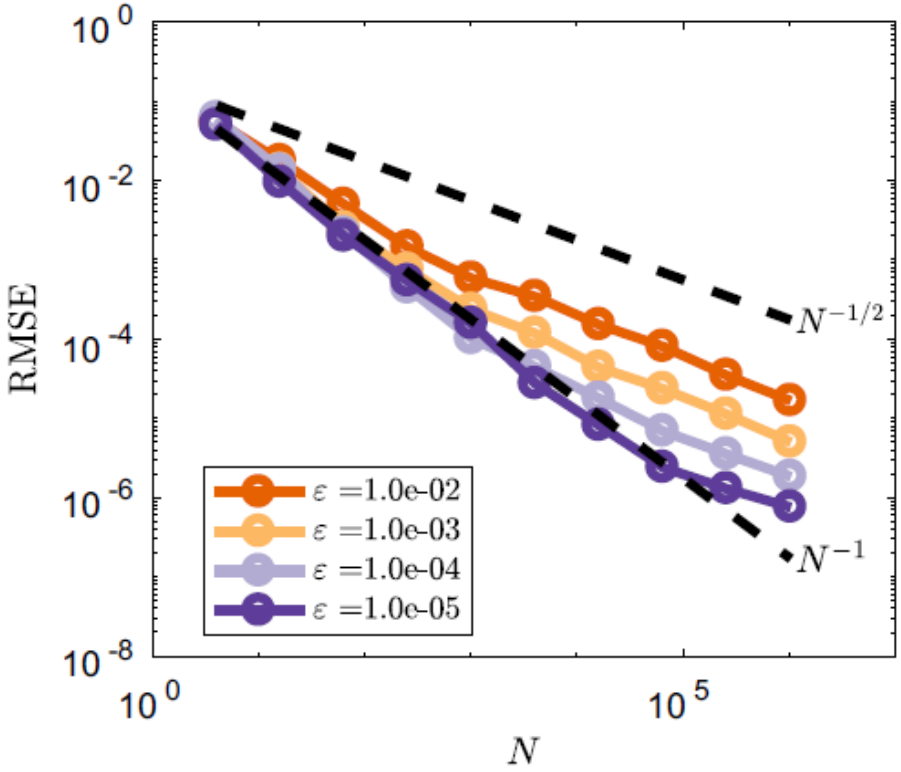

Figure: Comparison of RMSE convergence for the RQMC τ-leap method for the approximation of $\mathbb{E}[X(t)]$ for different values of ε.

Numerical results

- To explain this behavior we consider the integration over $[0,1)^s$ of

$$f(x) = \sqrt{\frac{12}{s}} \sum_{i=1}^{s} (x_i - \frac{1}{2})$$

- In this case, the RQMC approach works very well, achieving a convergence rate of $\mathcal{O}(N^{-\frac{3}{2}})$

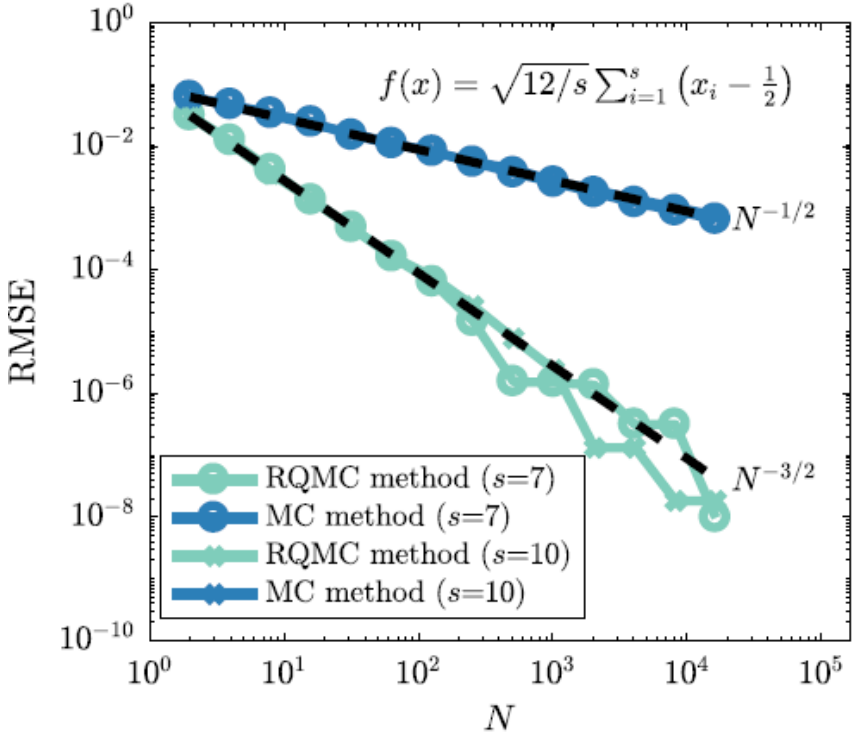

Figure: RMSE convergence rate of MC and RQMC for the integration of $f(x)$

Numerical results

- If we apply the transformation

$$f_\varepsilon(x) = \varepsilon \lfloor \frac{f(x)}{\varepsilon} \rfloor$$

the behavior mimics that of the RQMC τ-leap method

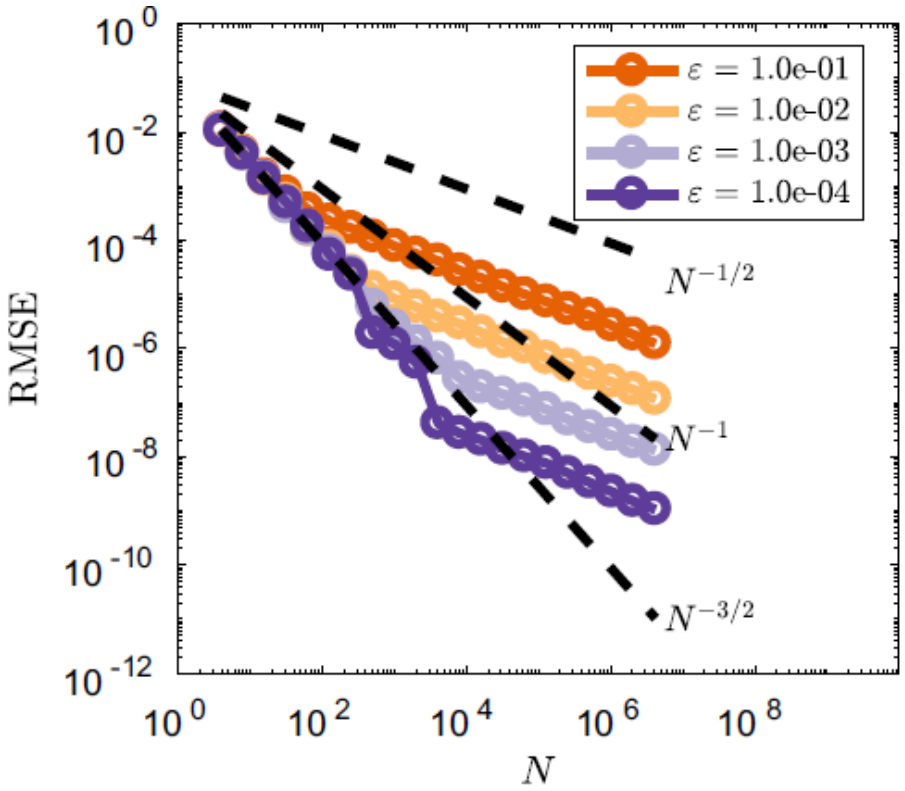

Figure: RMSE convergence rate for the function $f_\varepsilon(x)$ for different values of ε

Numerical results

- To explain this behavior, consider that we can split f_ε in the following way

$$f_\varepsilon(x) = \underbrace{f(x)}_{\text{continuous}} + \underbrace{(f_\varepsilon(x) - f(x))}_{\text{discontinuous}}$$

- Observe that $|f_\varepsilon(x) - f(x)| \leq \varepsilon$ and thus $Var(f_\varepsilon(x) - f(x)) = \mathcal{O}(\varepsilon^2)$
- Thus, we can split the MSE of the estimator in the following way

$$MSE(\varepsilon \lfloor \frac{f(x)}{\varepsilon} \rfloor) = C_1 N^{-3} + C_2 \varepsilon^2 N^{-1}$$

- It follows that the convergence rate of the MSE switches from $\mathcal{O}(N^{-3})$ to $\mathcal{O}(N^{-1})$ for $N = \mathcal{O}(\varepsilon^{-1})$, as the error for the discontinuous part of the function dominates the error for the continuous part

Numerical results

- For chemical reaction networks it is in general not clear how to split the quantity of interest into a continuous and discontinuous part
- As such, it is hard to estimate the benefit of using RQMC methods a priori

Numerical results

- To illustrate another disadvantage of using the RQMC τ-leap approach, we consider the Schlögl system, given by

$$\begin{aligned} 2S_1 + S_2 &\xrightarrow{c_1} 3S_1 \\ 3S_1 &\xrightarrow{c_2} 2S_1 + S_2 \\ S_3 &\xrightarrow{c_3} S_1 \\ S_1 &\xrightarrow{c_4} S_3 \end{aligned}$$

- We consider this system until final time $T = 4$ with a time step of $\tau = 0.4$, leading to a system dimension of 40

Numerical results

- Here, the RQMC methods for both the τ-leap and CLE approaches do not perform better than their standard MC counterparts
- This is likely do to the high dimensionality of the system

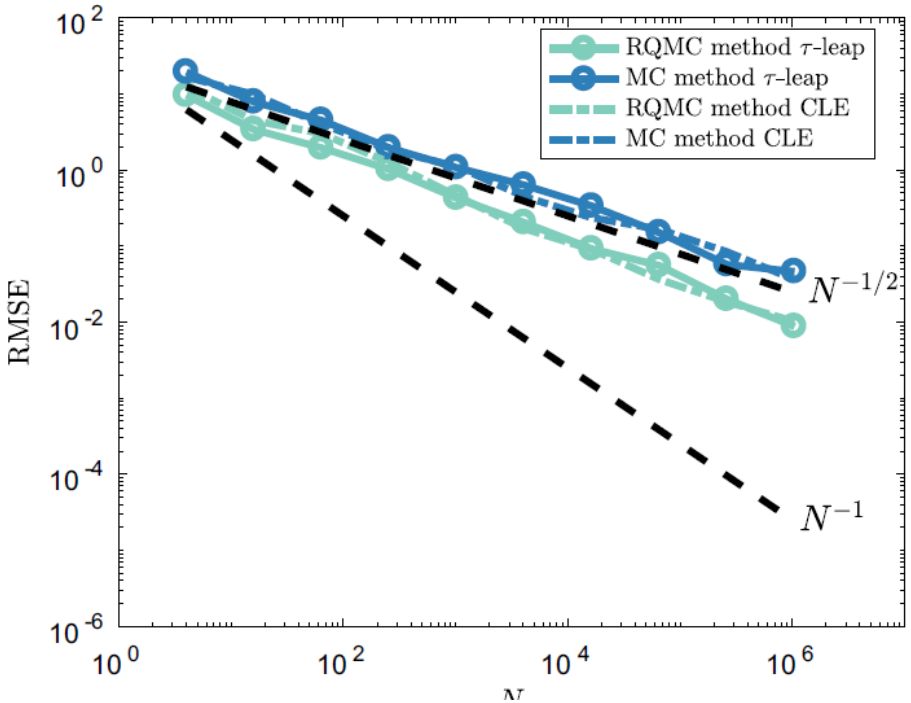

Figure: RMSE convergence for the mean number of S_1 molecules in the Schlögl system

Array-RQMC

- One approach to reducing the dimensionality of the problem is using the Array-RQMC method
- Instead of using $K\lceil \frac{T}{\tau} \rceil$-dimensional low discrepancy points, we instead use $l + K$-dimensional low-discrepancy points for some $l \in \{1, \ldots, n\}$ (Recall that n denotes the number of chemical species)
- These points are then used to advance all chains by one step at a time, after which they are randomized and sorted using an l-dimensional sort

Array-RQMC

- Specifically, we start with a low-discrepancy point set

$$\mathcal{Q}_N = \{(w_i, u_i), i = 0, \ldots, N-1\}$$

where $w_i \in [0,1)^l$ and $u_i \in [0,1)^K$, that is sorted with respect to the first l coordinates

- At each time step j, the last K coordinates of the points are randomized to obtain the RQMC point set

$$\tilde{\mathcal{Q}}_{N,j} = \{(w_i, U_{i,j}), i = 0, \ldots, N-1\}$$

where $U_{i,j} \sim \mathcal{U}([0,1)^K$

Array-RQMC

- Additionally, the states $X_{0,j-1}, \ldots, X_{N-1,j-1}$ are sorted with respect to their values $h(X_{0,j-1}), \ldots, h(X_{N-1,j-1})$ for a suitable mapping $h : \mathbb{N}^n \to \mathbb{R}^l$ using the same sorting algorithm as before
- Denoting the resulting permutation by π_j, the chains advance to step j via

$$X_{i,j} = X_{\pi_j(i),j-1} + \sum_{k=1}^{K} F^{-1}(U_{i,j,k}, a_k(X_{\pi_j(i),j-1})\tau)\zeta_k$$

Numerical results

- We again consider the Schlögl system with the same parameters as before, except for a time step size of $\tau = 0.25$, resulting in 16 timesteps
- We also make the simplifying assumption that only the copy number of S_1 changes
- This results in a dimensionality of 5 for the Array-RQMC approach, compared to a dimensionality of 64 for the regular RQMC approach

Numerical results

- VRF19 denotes the estimated variance reduction factor for $N = 2^{19}$, given

$$\frac{Var(\text{MC})}{Var([\text{Array}-]\text{RQMC})}$$

- EIF19 denotes the efficiency ratio for $N = 2^{19}$, given by

$$VRF19\frac{\text{CPU-time(MC)}}{\text{CPU-time}([\text{Array}-]\text{RQMC})}$$

- In all cases, Array-RQMC performs significantly better than regular RQMC

	$g(\mathbf{X}(t)) = X_1(t)$		
$(T, s, \tau) \longrightarrow$	(4, 16, 1/4)		
$\mathbb{E}[g(\mathbf{X}_s)]$	309.0		
MC Var	44,575		
Point sets	$\hat{\beta}$	VRF19	EIF19
MC	1.00	1	1
RQMC	1.10	9	9
Lat+s	**1.64**	2,897	2,458
Lat+s+b	1.24	10,147	9,318
Sob+LMS	1.56	**15,043**	**14,079**

Figure: Comparison of numerical results for the Schlögl system of regular Monte Carlo (MC), regular RQMC with Sobol points and a left random matrix scramble (RQMC), randomly shifted rank 1 lattice rules (Lat+s), Lat + s combined with the baker transformation (Lat+s+b) and a Sobol net with a random digital shift (Sob+LMS).

Heuristic argument for Array-RQMC

- Suppose that $l = n$ and that X_j is uniform over $[0,1)^n$ for each j
- At step j, the algorithm estimates for any function $g_j : [0,1)^n \to \mathbb{R}$

$$\mathbb{E}[g_j(X_j)] = \mathbb{E}[g_j(\varphi(X_{j-1}, U))] = \int_{[0,1)^{n+K}} g_j(\varphi(x,u))dxdu$$

by

$$\frac{1}{N}\sum_{i=0}^{N-1} g_j(X_{i,j}) = \frac{1}{N}\sum_{i=0}^{N-1} g_j(\varphi(X_{i,j-1}, U_{i,j}))$$

where φ is the transition function of the τ-leap algorithm

Heuristic argument for Array-RQMC

- This is an RQMC estimate with the point set

$$\mathcal{Q}_{N,j} = \{(X_{i,j-1}, U_{i,j}), i = 0, \ldots, N-1\}$$

- The goal is to have $\mathcal{Q}_{n,j}$ highly uniform over $[0,1)^{n+K}$
- As $X_{i,j-1}$ are dependent on the simulation, we order them in a way that $X_{i,j-1}$ is close to w_i for each i