Multilevel Drift-Implicit Tau-Leap

Chiheb Ben Hammouda, Alvaro Moraes and Raul Tempone

Computer, Electrical and Mathematical Science & Engineering Division
King Abdullah University of Science and Technology (KAUST), Saudi Arabia
{chiheb.benhammouda, alvaro.moraesgutierrez, raul.tempone}@kaust.edu.sa

Abstract

The dynamics of biochemical reactive systems with small copy numbers of one or more reactant molecules is dominated by stochastic effects. For those systems, discrete state-space and stochastic simulation approaches were proved to be more relevant than continuous state-space and deterministic ones. In systems characterized by having simultaneously fast and slow timescales, the existing discrete space-state stochastic path simulation methods such as the stochastic simulation algorithm (SSA) and the explicit tau-leap method can be very slow. Implicit approximations were developed in the literature to improve numerical stability and provide efficient simulation algorithms for those systems. In this work, we propose an efficient Multilevel Monte Carlo method in the spirit of the work by Anderson and Higham (2012) that uses drift-implicit tau-leap approximations at levels where the explicit tau-leap method is not applicable due to numerical stability issues. We present numerical examples that illustrate the performance of the proposed method.

Statement of the Problem

Let $\mathbf{X}$ be a Pure Jump Process

$$\mathbf{X} = (X_1, \ldots, X_d) : [0,T] \times \Omega \to \mathbb{Z}_+^d$$

described by

- Finite number of J possible reactions, defined by a stoichiometric vector (state change vector), $\boldsymbol{\nu}_j \in \mathbb{Z}$, such that for $\mathbf{x} \in \mathbb{Z}_+^d$
$$\mathbf{x} \to \mathbf{x} + \boldsymbol{\nu}_j, \quad j = 1 \ldots, J$$
- Propensity (jump intensity) functions, $a_j : \mathbb{R}_+^d \to \mathbb{R}_+$ defined by
$$\mathrm{P}\left(\mathbf{X}(t+\Delta t) = \mathbf{x} + \boldsymbol{\nu}_j \,\middle|\, \mathbf{X}(t) = \mathbf{x}\right) = a_j(\mathbf{x})\Delta t + o(\Delta t), \; j = 1, \ldots, J. \quad (1)$$

In this setting i-th component, $X_i(t)$, describes the abundance of the i-th species present in the chemical system at time t.

Goal: Given i) an initial state $\mathbf{X}_0 = \mathbf{x}_0$, ii) a smooth scalar observable of $\mathbf{X}$, $g : \mathbb{R}^d \to \mathbb{R}$, iii) a user-selected tolerance, TOL, and iv) a confidence level $1-\alpha$ close to 1, provide accurate estimator $\hat{Q}$ of $\mathrm{E}[g(\mathbf{X}(T))]$ such that

$$\mathrm{P}\left(\left|\mathrm{E}[g(\mathbf{X}(T))] - \hat{Q}\right| < TOL\right) > 1-\alpha, \quad (2)$$

with near-optimal expected computational work and for a class of systems characterized by having simultaneously fast and slow time scales (Stiff systems).

The process $\mathbf{X}$ can be characterized by the random time change representation

$$\mathbf{X}(t) = \mathbf{x}_0 + \sum_{j=1}^{J} Y_j\left(\int_0^t a_j(\mathbf{X}(s))\,\mathrm{d}s\right)\boldsymbol{\nu}_j, \quad (3)$$

where $Y_j : \mathbb{R}_+ \times \Omega \to \mathbb{Z}_+$ are independent unit-rate Poisson processes [4].

Idea: Multilevel Drift-Implicit Tau-leap

- **Multilevel Monte Carlo (MLMC)** Use a control variate multilevel approach to reduce the computational work (runtime) of the standard Monte Carlo method.
- **Coupling Drift-Implicit Tau-leap** Our MLMC strategy couples two drift-implicit tau-leap paths at the coarser levels of discretization until a certain interface level, L^{int} (defined by the numerical stability of the explicit tau-leap), where we start coupling paths using the explicit tau-leap method as indicated in [1]. In that sense, our strategy can be considered to be a hybrid algorithm. This strategy is specially relevant when TOL is small, implying that the finest level of the drift-implicit MLMC tau leap, L^{imp}, is in the stability regime of the explicit MLMC tau-leap. For large values of TOL, our MLMC estimator reduces to a pure drift-implicit MLMC tau-leap estimator.

Drift-Implicit Tau-leap Approximation

We define $\mathbf{Z}^{imp}$ and $\mathbf{Z}^{exp}$, the drift-implicit and the explicit tau-leap approximations of the process, $\mathbf{X}$, respectively.

The explicit tau-leap scheme, where $\mathbf{z} = \mathbf{Z}^{exp}(t)$, can be rewritten as follows:

$$\begin{aligned}\mathbf{Z}^{exp}(t+\tau) &= \mathbf{z} + \sum_{j=1}^{J} \mathcal{P}_j\left(a_j(\mathbf{z})\tau\right)\boldsymbol{\nu}_j \\ &= \mathbf{z} + \sum_{j=1}^{J} a_j(\mathbf{z})\tau\boldsymbol{\nu}_j + \sum_{j=1}^{J}\left(\mathcal{P}_j\left(a_j(\mathbf{z})\tau\right) - a_j(\mathbf{z})\tau\right)\boldsymbol{\nu}_j. \end{aligned} \quad (4)$$

Let us denote the second and third quantities in the right-hand side of (4) by the drift and the zero-mean noise, respectively. The idea of drift-implicit tau-leap methods is to take only the drift part as implicit while the noise part is left explicit. Defining $\mathbf{z} = \mathbf{Z}^{imp}(t)$, then a drift -implicit tau-leap scheme consists of the following two steps:

$$\begin{aligned}\mathbf{y} &= \mathbf{z} + \sum_{j=1}^{J} a_j\left(\mathbf{y}\right)\tau\boldsymbol{\nu}_j \text{ (Drift-Implict step)} \quad (5)\\ \mathbf{Z}^{imp}(t+\tau) &= \mathbf{y} + \sum_{j=1}^{J}\left(\mathcal{P}_j(a_j(\mathbf{y})\tau) - a_j(\mathbf{y})\tau\right)\boldsymbol{\nu}_j \\ &= \mathbf{z} + \sum_{j=1}^{J}\mathcal{P}_j(a_j(\mathbf{y})\tau)\boldsymbol{\nu}_j \text{ (Tau-leap step)}\end{aligned}$$

Notice that the scheme that we propose naturally produces values of $\mathbf{Z}^{imp}$ in $\mathbb{Z}_+^d$. Consequently, we do not need rounding as suggested in [7].

Multilevel Monte Carlo

Multilevel by Giles [5]: Reduce the variance of the standard Monte Carlo method.

- A hierarchy of nested meshes of the time interval $[0,T]$, indexed by $\ell = 0, 1, \ldots, L$.
- $h_\ell = M^{-\ell}h_0$: The size of the subsequent time steps for levels $\ell \geq 1$, where $M>1$ is a given integer constant and h_0 the step size used at level $\ell = 0$.
- $\mathbf{Z}_\ell$: The approximate process generated using a step size of h_ℓ.

Consider now the following telescoping decomposition of $\mathrm{E}[g(\mathbf{Z}_L(T))]$:

$$\mathrm{E}[g(\mathbf{Z}_L(T))] = \mathrm{E}[g(\mathbf{Z}_0(T))] + \sum_{\ell=1}^{L}\mathrm{E}[g(\mathbf{Z}_\ell(T)) - g(\mathbf{Z}_{\ell-1}(T))]. \quad (6)$$

By defining

$$\begin{cases}\hat{Q}_0 := \frac{1}{N_0}\sum_{n_0=1}^{N_0} g(\mathbf{Z}_{0,[n_0]}(T)) \\ \hat{Q}_\ell := \frac{1}{N_\ell}\sum_{n_\ell=1}^{N_\ell}\left(g(\mathbf{Z}_{\ell,[n_\ell]}(T)) - g(\mathbf{Z}_{\ell-1,[n_\ell]}(T))\right),\end{cases} \quad (7)$$

we arrive at the unbiased MLMC estimator, $\hat{Q}$, of $\mathrm{E}[g(\mathbf{Z}_L(T))]$:

$$\hat{Q} := \sum_{\ell=0}^{L}\hat{Q}_\ell. \quad (8)$$

Coupling Two Drift-Implicit Tau-Leap Paths

- Extending MLMC to the setting of Stochastic Reaction Networks requires simulating the paths $\mathbf{Z}_l$ and $\mathbf{Z}_{l-1}$ simultaneously in an efficient manner that produces small variances between the paths. We extend the idea of coupling explicit tau-leap paths introduced in [1] to define the multilevel drift-implicit tau-leap estimator.
- This idea is based on coupling two Poisson random variables, $\mathcal{P}_1(\lambda_1)$, $\mathcal{P}_2(\lambda_2)$, with rates λ_1 and λ_2, respectively, by defining $\lambda^\star := \min\{\lambda_1, \lambda_2\}$ and considering the decomposition

$$\begin{cases}\mathcal{P}_1(\lambda_1) := \mathcal{Q}(\lambda^\star) + \mathcal{Q}_1(\lambda_1 - \lambda^\star) \\ \mathcal{P}_2(\lambda_2) := \mathcal{Q}(\lambda^\star) + \mathcal{Q}_2(\lambda_2 - \lambda^\star)\end{cases} \quad (9)$$

where $\mathcal{Q}(\lambda^\star)$, $\mathcal{Q}_1(\lambda_1 - \lambda^\star)$ and $\mathcal{Q}_2(\lambda_2 - \lambda^\star)$ are three independent Poisson random variables.

Multilevel Drift-Implicit Tau-leap Estimator

Let us consider a family of uniform time meshes of $[0,T]$, with size $h_\ell = 2^{-\ell}T$. Let L_c^{imp} and L_c^{exp} be the coarsest levels in which the drift-implicit and the explicit methods are respectively numerical stable. In the class of problems we are interested in, we have the relation $h_{L_c^{\text{exp}}} \ll h_{L_c^{\text{imp}}}$, which means that L_c^{imp} is much coarser than L_c^{exp}.

Rewriting (8) in our context, we define our drift-implicit MLMC tau-leap estimator as

$$\hat{Q} := \hat{Q}_{L_c^{\text{imp}}} + \sum_{\ell=L_c^{\text{imp}}+1}^{L^{\text{int}}-1}\hat{Q}_\ell + \hat{Q}_{L^{\text{int}}} + \sum_{\ell=L^{\text{int}}+1}^{L}\hat{Q}_\ell, \quad (10)$$

where

$$\begin{cases}\hat{Q}_{L_c^{\text{imp}}} := \frac{1}{N_{i,L_c^{\text{imp}}}}\sum_{n=1}^{N_{i,L_c^{\text{imp}}}} g(\mathbf{Z}^{imp}_{L_c^{\text{imp}},[n]}(T)) \\ \hat{Q}_\ell := \frac{1}{N_{ii,\ell}}\sum_{n_\ell=1}^{N_{ii,\ell}}\left(g(\mathbf{Z}^{imp}_{\ell,[n_\ell]}(T)) - g(\mathbf{Z}^{imp}_{\ell-1,[n_\ell]}(T))\right), \quad L_c^{\text{imp}}+1 \leq \ell \leq L^{\text{int}}-1 \\ \hat{Q}_{L^{\text{int}}} := \frac{1}{N_{ie,L^{\text{int}}}}\sum_{n=1}^{N_{ie,L^{\text{int}}}}\left(g(\mathbf{Z}^{exp}_{L^{\text{int}},[n]}(T)) - g(\mathbf{Z}^{imp}_{L^{\text{int}}-1,[n]}(T))\right) \\ \hat{Q}_\ell := \frac{1}{N_{ee,\ell}}\sum_{n_\ell=1}^{N_{ee,\ell}}\left(g(\mathbf{Z}^{exp}_{\ell,[n_\ell]}(T)) - g(\mathbf{Z}^{exp}_{\ell-1,[n_\ell]}(T))\right), \quad L^{\text{int}}+1 \leq \ell \leq L.\end{cases} \quad (11)$$

- L_c^{imp}, the coarsest discretization level.
- L^{int}, the interface level.
- L, the finest level of discretization.
- $\mathbf{N} := \{N_{i,L_c^{\text{imp}}}, \{N_{ii,\ell}\}_{\ell=L_c^{\text{imp}}+1}^{L^{\text{int}}-1}, N_{ie,L^{\text{int}}}, \{N_{ee,\ell}\}_{\ell=L^{\text{int}}+1}^{L}\}$, the number of samples per level.

Estimation procedure

- **Coarsest discretization level**, L_c^{imp}, is determined by the numerical stability constraint of our MLMC estimator, two conditions must be satisfied:
  - The first one ensures the stability of a single path, which is related to the coarsest level of discretization, L_c, and which can be determined by a linearized stability analysis of the backward Euler method applied to the deterministic ODE model corresponding to our system [7].
  - The second one ensures the stability of the variance of the coupled paths of our MLMC estimator and can be expressed by $\mathrm{Var}\left[g(\mathbf{Z}_{L_c^{\text{imp}}+1}) - g(\mathbf{Z}_{L_c^{\text{imp}}})\right] \ll \mathrm{Var}\left[g(\mathbf{Z}_{L_c^{\text{imp}}})\right]$.
- **Total number of levels**, L, and the set of **the number of samples per level**, $\mathbf{N}$, are selected to satisfy the accuracy constraint given by (2) (typically we choose $\alpha = 0.05$), with near-optimal expected computational work. As a result, the MLMC algorithm should bound the bias and the statistical error as follows:

$$\left|\mathrm{E}\left[g(\mathbf{X}(T)) - \hat{Q}\right]\right| \leq (1-\theta)\,TOL, \quad (12)$$

$$\mathrm{Var}\left[\hat{Q}\right] \leq \left(\frac{\theta\,TOL}{C_\alpha}\right)^2 \quad (13)$$

for some given confidence parameter, C_α, such that $\Phi(C_\alpha) = 1 - \alpha/2$; here, Φ is the cumulative distribution function of a standard normal random variable (see [3] for details).

In our problem, the finest discretization level, L, is determined by satisfying relation (12) for $\theta = \frac{1}{2}$, implying

$$|\text{Bias}(L) := \mathrm{E}[g(\mathbf{X}(T)) - g(\mathbf{Z}_L(T))]| < \frac{TOL}{2}. \quad (14)$$

In our numerical experiments, we use the following approximation (see [5])

$$\text{Bias}(L) \approx \mathrm{E}[g(\mathbf{Z}_L(T)) - g(\mathbf{Z}_{L-1}(T))]. \quad (15)$$

Therefore, to determine the value of L, we need to have estimates of the bias for different levels of discretization, ℓ.

- **Interface level**, L^{int} and **optimal number of samples per level**, $\mathbf{N}$:

Given L, we define $W_{L^{\text{int}}}$ to be the expected computational cost of the MLMC estimator given that the interface level is L^{int}

$$\begin{aligned}W_{L^{\text{int}}} := {} & C_{i,L_c^{\text{imp}}}N_{i,L_c^{\text{imp}}}h^{-1}_{L_c^{\text{imp}}} + \sum_{\ell=L_c^{\text{imp}}+1}^{L^{\text{int}}-1} C_{ii,\ell}N_{ii,\ell}h_\ell^{-1} + C_{ie,L^{\text{int}}}N_{ie,L^{\text{int}}}h^{-1}_{L^{\text{int}}} \\ & + \sum_{\ell=L^{\text{int}}+1}^{L} C_{ee,\ell}N_{ee,\ell}h_\ell^{-1}, \end{aligned} \quad (16)$$

where C_i, C_{ii}, C_{ie} and C_{ee} are, respectively, the expected computational costs of simulating a single drift-implicit tau-leap step, a coupled drift-implicit tau-leap step, a coupled drift-implicit/explicit tau-leap step and a coupled explicit tau-leap step.

1. The first step is to solve (17), for a fixed value of the interface level, L^{int}

$$\begin{cases}\min_{\mathbf{N}} W_{L^{\text{int}}}(\mathbf{N}) \\ s.t.\; C_\alpha\sqrt{\sum_{\ell=L_c^{\text{imp}}}^{L} N_\ell^{-1}V_\ell} \leq \frac{TOL}{2},\end{cases} \quad (17)$$

where $V_\ell = \mathrm{Var}\left[g(\mathbf{Z}_\ell(T)) - g(\mathbf{Z}_{\ell-1}(T)\right]$ is estimated, like the bias, by the extrapolation of the sample variances obtained from the coarsest levels (due to the presence of large kurtosis (see Section 1 of [6]) problem).

2. Let us denote $\mathbf{N}^*(L^{\text{int}})$ as the solution of (17). Then, the optimal value of the switching parameter, L^{int}, should be chosen to minimize the expected computational work; that is, the value $L^{\text{int}*}$ that solves

$$\begin{cases}\min_{L^{\text{int}}} W_{L^{\text{int}}}(\mathbf{N}^*(L^{\text{int}})) \\ s.t.\; L_c^{\text{exp}} \leq L^{\text{int}} \leq L.\end{cases}$$

In our numerical examples, we found that the lowest computational cost is achieved for $L^{\text{int}*} = L_c^{\text{exp}}$, *i.e.*, the same level in which the explicit tau-leap is stable.

Numerical Example

This example was studied in [7] and is given by the following reaction set

$$\begin{aligned} S_1 &\xrightarrow{c_1} S_3, \\ S_3 &\xrightarrow{c_2} S_1 \\ S_1 + S_2 &\xrightarrow{c_3} S_1 + S_4. \end{aligned} \quad (18)$$

Since the total number of S_1 and S_3 molecules is constant (say K), and if we ignore the by-product, S_4, this system can be represented by three reactions and two variables $\mathbf{X} = (X_1, X_2)$, which are numbers of S_1 and S_2 molecules, respectively. The stoichiometric vectors are $\boldsymbol{\nu}_1 = (-1,0)^T$, $\boldsymbol{\nu}_2 = (1,0)^T$, and $\boldsymbol{\nu}_3 = (0,-1)^T$ and the corresponding propensity functions are

$$a_1(\mathbf{X}) = c_1X_1, \; a_2(\mathbf{X}) = c_2(K - X_1), \; a_3(\mathbf{X}) = c_3X_1X_2. \quad (19)$$

We chose the same values for the different parameters as in [7]: $c_1 = c_2 = 10^5$, $c_3 = 5 \times 10^{-3}$, $K = 2 \times 10^4$ and initial condition $\mathbf{X}(0) = (10^4, 10^2)^T$. This setting implies that the stability limit of the explicit tau-leap is $\tau_{\text{exp}}^{\text{lim}} \approx 10^{-5}$. We consider the final time, $T = 0.01$, seconds. In the following numerical experiments, we are interested in approximating $\mathrm{E}[X_2(T)]$.

Method / TOL	0.02	0.01	0.005
Explicit MLMC ($L_c^{\text{exp}} = 11$)	8.9e+02 (9)	5.3e+03 (45)	2.2e+04 (96)
Drift-Implicit MLMC ($L_c^{\text{imp}} = 0$)	2.4e+01 (0.8)	1.1e+02 (3)	5.3e+02 (7)
$W^{exp}_{MLMC}/W^{imp}_{MLMC}$	37.04	47.33	41.27

Table 1: *Comparison of the expected total work for the different methods (in seconds) using 100 multilevel runs. The quantity between parenthesis,* i.e., $(\cdot)$*, refers to the standard deviation.*

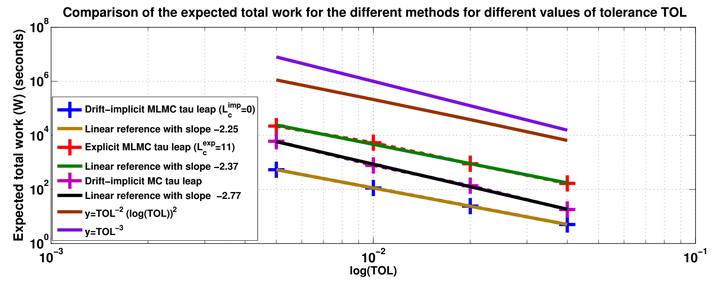

Figure 1: *Comparison of the expected total work for the different methods with different values of tolerance (TOL) using 100 multilevel runs.*

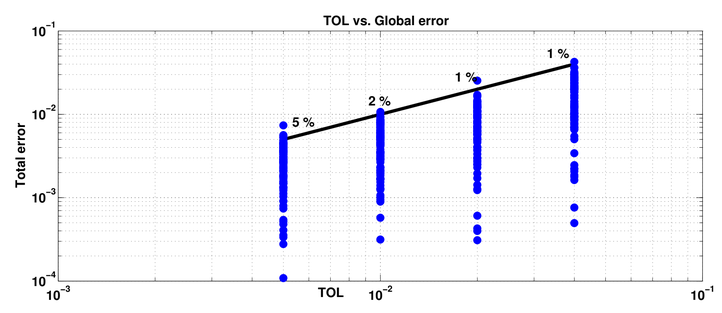

Figure 2: *TOL versus the actual computational error. The numbers above the straight line show the percentage of runs that had errors larger than the required tolerance.*

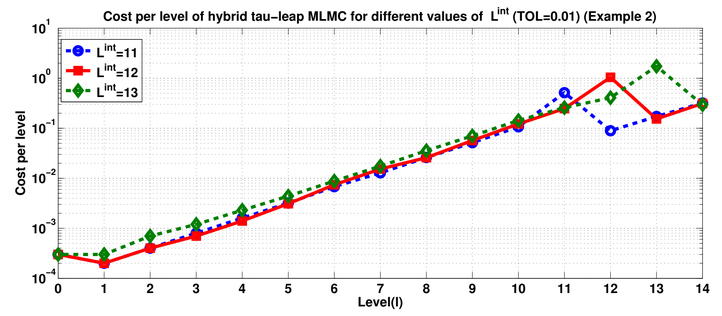

Figure 3: *Cost per level of the drift-implicit MLMC tau-leap estimator for different values of L^{int} ($TOL = 0.01$).*

Conclusions

- Our proposed estimator is useful in systems with the presence of slow and fast timescales (stiff systems).
- Through our numerical experiments, we obtained substantial gains with respect to both the explicit MLMC and the drift-implicit, single-level tau-leap methods. We also showed that for large values of TOL the pure drift-implicit MLMC method has the same order of computational work as does the explicit MLMC tau-leap methods, which is of $\mathcal{O}\left(TOL^{-2}\log(TOL)^2\right)$ [2], but with a smaller constant.

Acknowledgements

The authors are members of the KAUST SRI Center for Uncertainty Quantification in Computational Science and Engineering.

References

[1] D. Anderson and D. Higham. Multilevel Monte Carlo for continuous Markov chains, with applications in biochemical kinetics. *SIAM Multiscal Model. Simul.*, 10(1), 2012.

[2] D. F. Anderson, D. J. Higham, and Y. Sun. Complexity of multilevel Monte Carlo tau-leaping. 52(6):3106–3127, 2014.

[3] N. Collier, A.-L. Haji-Ali, F. Nobile, E. von Schwerin, and R. Tempone. A continuation multilevel monte carlo algorithm. *BIT Numerical Mathematics*, 55(2):399–432, 2014.

[4] S. N. Ethier and T. G. Kurtz. *Markov Processes: Characterization and Convergence (Wiley Series in Probability and Statistics)*. Wiley-Interscience, 2nd edition, 9 2005.

[5] M. Giles. Multi-level Monte Carlo path simulation. *Operations Research*, 53(3):607–617, 2008.

[6] A. Moraes, R. Tempone, and P. Vilanova. Multilevel hybrid chernoff tau-leap. *BIT Numerical Mathematics*, pages 1–51, 2015.

[7] M. Rathinam, L. Petzold, Y. Cao, and D. T. Gillespie. Stiffness in stochastic chemically reacting systems: the implicit tau-leaping method. *Journal of Chemical Physics*, 119(24):12784–12794, Dec 2003.