Hybrid Multilevel Monte Carlo Simulation of Stochastic Reaction Networks

Alvaro Moraes
joint work with Raúl Tempone and Pedro Vilanova
http://sri-uq.kaust.edu.sa

Thuwal, KSA, Jan. 6, 2015
Our Contribution

This presentation is based on:


Part I

1. State the problem
2. Exact and approximate simulation of Stochastic Reaction Networks
3. Chernoff Tau-leap
4. Hybrid paths
5. Global error control
6. Work optimization strategies
7. Some numerical results
Statement of the problem

Let $X$ be a **Stochastic Reaction Network**

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

described by $J$ reaction channels, $R_j := (\nu_j, a_j)$, where

- $\nu_j \in \mathbb{Z}^d$ is called **stoichiometric vector**
- $a_j : \mathbb{Z}_+^d \rightarrow \mathbb{R}^+$ is called **propensity function**

such that

$$P \left( X(t + dt) = x + \nu_j \mid X(t) = x \right) = a_j(x)dt + o(dt).$$

**Goal:** Given an observable $g : \mathbb{Z}_+^d \rightarrow \mathbb{R}$, a tolerance $TOL > 0$, and a small number $\alpha > 0$, find an estimator $\mathcal{M}$ of $E[g(X(T))]$ such that

$$P \left( |E[g(X(T))] - \mathcal{M}| > TOL \right) < \alpha$$

with near-optimal computational work.
Total error \( |E[g(X(T))] - M| \) vs \( TOL \)

Figure: \( P\left( |E[g(X(T))] - M| > TOL \right) < \alpha = 5\% \)
Example: Gene transcription and translation

- $\emptyset \xrightarrow{25} M$, a gene is being transcribed into mRNA.
- $M \xrightarrow{1000} M + P$, mRNA is then being translated into proteins.
- $P + P \xrightarrow{0.001} D$, finally the proteins produce stable Dimers.
- $M \xrightarrow{0.1} \emptyset$, $P \xrightarrow{1} \emptyset$ degradation of mRNA and proteins, resp.

Estimate the expected number of Dimers at time $T=1$ up to certain tolerance $TOL$, with high probability.

$$\nu = (\nu_j)_{j=1}^J := \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -2 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix}^T$$

and

$$a(X) := \begin{pmatrix} 25 \\ 10^3 M \\ 10^{-3} P(P-1) \\ 0.1 M \\ P \end{pmatrix}$$

where $X(t) = (M(t), P(t), D(t))$. 
Example: Gene transcription and translation

Figure: Exact paths of the time evolution of the species in the gene transcription and translation (GTT) example.
The Stochastic Simulation Algorithm (SSA)[Gillespie, 1976]

It produces statistically correct path-simulations of $X$. Remember that $X$ is a continuous-time Markov chain living in $\mathbb{Z}_+^d$.

1. In state $x$ at time $t$, compute $(a_j(x))_{j=1}^J$ and their sum $a_0(x) = \sum_{1 \leq j \leq J} a_j(x)$.

2. Simulate the time to the next reaction, $\tau$, as an exponential random variable with rate $a_0(x)$.
   Observe: $E[\tau] = (a_0(x))^{-1}$ could be very small!.

3. Simulate independently the next reaction, $\nu_j$, according to the probability mass function values $(a_j(x)/a_0(x))_{j=1}^J$.

4. Update: $t \leftarrow t + \tau$ and $x \leftarrow x + \nu_j$.

5. Record $(t, x)$. Return to step one if $t < T$, otherwise end the simulation.
The Tau-leap method (TL)  
[Aparicio and Solari, 2001, Gillespie, 2001]

From Kurtz’s random time change representation [Kurtz, 1978],

\[ X(t) = X(0) + \sum_{j=1}^{J} Y_j \left( \int_{0}^{t} a_j(X(s))ds \right) \nu_j, \]

where \( Y_j \) are independent unit-rate Poisson processes, we obtain the Tau-leap method (kind of forward Euler approximation):

Given \( \bar{X}(t) = x \in \mathbb{Z}_+^d \),

\[ \bar{X}(t+\tau) = x + \sum_{j=1}^{J} \mathcal{P}_j \begin{pmatrix} a_j(x) \tau \end{pmatrix} \nu_j, \]

where \( \mathcal{P}_j(\lambda_j) \) are independent Poisson random variables with rate \( \lambda_j \). Danger: \( \bar{X}(t+\tau) \) may have some negative components!
A Chernoff bound for the Tau-leap method I

Problem: Given $\delta > 0$, find the largest $\tau = \tau(x, \delta)$ s.t.

$$P \left( \bar{X}(t+\tau) \notin \mathbb{Z}_+^d \mid \bar{X}(t) = x \right) \leq ChBnd(x, \tau) < \delta.$$ 

Idea: Use the moment generating function of a linear combination of independent Poisson random variables to produce a Chernoff bound $ChBnd(x, \tau)$. For $i = 1, \ldots, d$, solve numerically

$$\sup_{\tau_i > 0} \inf_{s > 0} \exp \left\{ -sx_i + \tau_i \sum_{j=1}^{J} a_j(x)(e^{-sv_{ji}} - 1) \right\} = \delta / d.$$ 

We define $\tau_{Ch} := \min\{\tau_1, \ldots, \tau_d\}$. 
What is a Chernoff bound?

**Single-reaction case**

Let $Q \sim \text{Poisson}(\lambda)$, $\lambda > 0$. Given $n \geq 0$ integer, the Chernoff bound is given by

$$P(Q \geq n) \leq \exp\left(n(1 - \log(n/\lambda) - \lambda)\right),$$

valid for $\lambda < n$ (otherwise the bound is trivial).

**Proof:** First note that for $s > 0$ the Markov inequality gives

$$P(Q \geq n) = P\left(e^{sQ} \geq e^{sn}\right) \leq \frac{E[e^{sQ}]}{e^{sn}},$$

$$P(Q \geq n) \leq \exp\left(\inf_{s > 0}\{-sn + \lambda(e^s - 1)\}\right).$$

When $\lambda \in (0, n)$, $\inf_{s > 0}\{-sn + \lambda(e^s - 1)\}$ is achieved at $\tilde{s} = \log(n/\lambda)$, and its value is $n(1 - \log(n/\lambda) - \lambda)$. 
Figure: Let $n = 10$ and $\lambda \in (2, 10)$. Semi-logarithmic plot of $P(Q(\lambda) \geq n) \leq ChBnd(n, \lambda) = \exp\left( n(1 - \log(n/\lambda) - \lambda) \right)$. See Klar’s bound in [Klar, 2000]. See also [Cao et al., 2005a]
A Hybrid Chernoff Tau-Leap Algorithm

Main issues:

• **Exact algorithms** like SSA and MNRM may be not computationally feasible, since the expected inter-arrival time between transitions, $\tau_{\text{SSA}}(x) = (a_0(x))^{-1}$, where $x = X(t)$ could be very small.

• **Approximate algorithms** that evolve with fixed time steps, like the Tau-leap, may be faster [Aparicio and Solari, 2001, Gillespie, 2001], but introduces discretization error and can jump outside $\mathbb{Z}_d^+$ (non physical results). Pre-leap: adjust the time step to control the one-step exit probability, possibly enforcing too small steps.

Main idea:

• We propose a **hybrid** algorithm that, at each time step, adaptively switches between SSA and Tau-leap.
One-step switching rule

Algorithm 1 From $\bar{X}(t) = x$ take one step. $T_0$ is the next grid point.

1: Compute $\tau_{SSA}$.
2: if $K_1 \tau_{SSA} > T_0 - t$ then
3: Use SSA
4: else
5: Compute $\tau_{Ch}$
6: if $\tau_{Ch} \geq K_2 \tau_{SSA}$ then
7: Use Tau-Leap
8: else
9: Use SSA
10: end if
11: end if

Note: $K_1$ is the cost of computing $\tau_{Ch}(x)$ divided by the cost of taking an SSA step. $K_2$ is the cost of taking a Chernoff Tau-leap step divided by the cost of taking an SSA step plus the cost of computing $\tau_{Ch}(x)$.
One-step switching rule in the GTT model

Figure: Regions of the one-step switching rule in the GTT model. The blue and red dots show the Chernoff tau-leap and the SSA regions, respectively. From left to right, $\delta = 10^{-2}, 10^{-4}, 10^{-6}$, respectively.
Hybrid Tau-leap algorithm

**Algorithm 2** Given \( \tilde{X}(t_0) = x_0 \), simulates a hybrid path.

1: while \( t < T \) do
2: \hspace{1em} method ← One-step rule
3: \hspace{1em} if method = SSA then
4: \hspace{2em} Apply SSA and advance one step
5: \hspace{1em} else
6: \hspace{2em} \( \tilde{X}_t' \leftarrow \tilde{X}_t + \mathcal{P}(a(\tilde{X}_t)TCh)\nu \)
7: \hspace{2em} if \( \tilde{X}_t' \notin \mathbb{Z}_+^d \) then
8: \hspace{3em} return
9: \hspace{2em} end if
10: \hspace{2em} \( \tilde{X}_t \leftarrow \tilde{X}_t' \)
11: \hspace{1em} \( t \leftarrow t + TCh \)
12: \hspace{1em} \( N_{TL} \leftarrow N_{TL} + 1 \)
13: \hspace{1em} end if
14: end while
An exiting path is a rare event: the role of $\delta$

Let $A$ be the event that a hybrid trajectory arrived to final time $T$ without exiting $\mathbb{Z}_+^d$. We show in Ref 1 that

$$P(A^c) \leq \delta E[N_{TL}] - \frac{\delta^2}{2} (E[N_{TL}^2] - E[N_{TL}]) + o(\delta^2).$$

In practice, we use $\delta E[N_{TL}]$ as an upper bound of $P(A^c)$. We also prove that $E[N_{TL}]$ is bounded for polynomial propensity functions and tends to zero when $\delta \to 0$.

Remark: The role of $\delta$ is to turn $A^c$ into a rare event. Direct sampling of hybrid paths to estimate $P(A^c)$ is non feasible, while the estimate of $E[N_{TL}]$ is straightforward.
Global Error Decomposition

Define the Monte Carlo estimator $\mathcal{M}$ as

$$\mathcal{M} := \frac{1}{M} \sum_{m=1}^{M} (g(\bar{X}(T))1_A) (\bar{\omega}_m).$$

Define the computational global error, $\mathcal{E}$, as

$$\mathcal{E} := E[g(X(T))] - \mathcal{M}.$$

Global Error decomposition (notation: $\bar{g} := g(\bar{X}(T))$)

$$\mathcal{E} = E[g(X(T))(1_A + 1_{A^c})] \pm E[\bar{g}1_A] - \mathcal{M}$$

$$= E[g(X(T))1_{A^c}] + E[(g(X(T)) - \bar{g})1_A] + E[\bar{g}1_A] - \mathcal{M}.$$

Problem: Given $\text{TOL} > 0$ and $\alpha > 0$, find the parameters for computing $\mathcal{M}$ such that $|\mathcal{E}| < \text{TOL}$ with confidence level $1 - \alpha$, and with nearly optimal computational work.
Estimation procedure

We have a 3-phase estimation procedure:

Phase 1 (off-line) Calibration of virtual machine-dependent quantities, for examples $C_i$, $i = 1, 2, 3$.

Phase 2 Given a prescribed TOL and a confidence level $\alpha$, find:

- $\delta$ (upper bound for the one-step exit probability for controlling the global exit error).
- a time mesh of size $h$ (for controlling the time discretization error).
- $M$ (the number of Monte Carlo realizations for controlling the statistical error).

such that the estimator $\mathcal{M}$ can be computed with near-optimal computational work.

Phase 3 Estimation of $\mathbb{E}[g(X(T))]$. 
About Phase 2, I

Define the expected cost of a hybrid trajectory,

\[
\Psi(h, \delta) := C_1 E[N_{SSA,K_1}(h, \delta)] + C_2 E[N_{SSA,K_2}(h, \delta)] + C_3 E[N_{TL}(h, \delta)] + \sum_{j=1}^{J} E \left[ \int_{[0,T]} C_P(a_j(\bar{X}(s))\tau_{Ch}(\bar{X}(s), \delta))1_{TL}(\bar{X}(s))ds \right],
\]

we want to find an approximate solution of

\[
\begin{cases}
\min_{M,h,\delta} M\Psi(h, \delta) \\
\text{s.t.} \\
\mathcal{E}_I + \mathcal{E}_E + \mathcal{E}_S \leq TOL
\end{cases}
\]

where \( C_1, C_2, C_3 \) are machine-dependent work quantities (Alg. 1) and \( C_P \) is the computational work of the Gamma simulation method for Poisson variates [Ahrens and Dieter, 1974],
Figure: Computational work of generating Poisson deviates (computer and MATLAB dependent).
About Phase 2, III

For a given $TOL > 0$, we refine $\delta$ until $E_E \leq TOL^2$,

$$\min_{M,h} M\Psi(h,\delta_0)$$

s.t.

$$\varepsilon_I(h,\delta_0) + \varepsilon_S \leq TOL - TOL^2$$

We now refine the mesh until $\varepsilon_I(h_0,\delta_0) + \varepsilon_S \leq TOL - TOL^2$, using an intermediate value for $M_{aux} = M_{aux}(h,\delta,\Psi)$, to finally get

$$M(h_0,\delta_0) = \left( \frac{C_A \sqrt{S^2 (g(X(T)); M_s)}}{TOL - TOL^2 - \varepsilon_I(h_0,\delta_0)} \right)^2.$$  

Finally, if $M\Psi < M_{SSA} E[N_{SSA^*}]$ use Hybrid, otherwise use SSA.
Figure: As the TOL goes to zero, the error bound goes to zero and the expected computational work of the Hybrid method tends to the expected work of the SSA. Observe the agreement between predicted and actual work. (1)
Part II

1. The Multilevel Monte Carlo idea
2. A result about complexity of our MLMC algorithm
3. Some numerical results
Introducing some notation

Consider a hierarchy of nested time meshes of the interval $[0, T]$, indexed by $\ell = 0, 1, \ldots, L$, such that

1. $\Delta t_0$ is the size of the coarsest time mesh.
2. $\Delta t_\ell = 2^{-\ell} \Delta t_0$, $\ell = 1, \ldots, L$.

Let

- $\tilde{X}_\ell(\cdot):=\tilde{X}(\cdot; \Delta t_\ell, \delta)$ be a hybrid Chernoff tau-leap path generated using a time mesh of size $\Delta t_\ell$ and one-step exit probability bound $\delta$.
- $A_\ell:=\{\tilde{\omega} \in \tilde{\Omega} : \tilde{X}_\ell(t) \in \mathbb{Z}_+^d, \forall t \in [0, T]\}$.
- $g_\ell := g(\tilde{X}_\ell(T))$. 

Consider the following telescoping decomposition (see Ref 2):

\[
E[g_L 1_{A_L}] = E[g_0 1_{A_0}] + \sum_{\ell=1}^{L} E[g_{\ell} 1_{A_{\ell}} - g_{\ell-1} 1_{A_{\ell-1}}],
\]

which motivates our MLMC estimator of \(E[g(X(T))]\),

\[
\mathcal{M}_L := \frac{1}{M_0} \sum_{m_0=1}^{M_0} g_0 1_{A_0}(\omega_{m_0}) + \sum_{\ell=1}^{L} \frac{1}{M_\ell} \sum_{m_\ell=1}^{M_\ell} [g_{\ell} 1_{A_{\ell}} - g_{\ell-1} 1_{A_{\ell-1}}](\omega_{m_\ell}).
\]

We define the computational global error, \(\mathcal{E}_L\), as

\[
\mathcal{E}_L := E[g(X(T))] - \mathcal{M}_L.
\]
The MLMC estimator and Global Error Decomposition II

Now, consider the following decomposition of $\mathcal{E}_L$

$$
\mathcal{E}_L = \mathbb{E} \left[ g(X(T))(1_{A_L} + 1_{A_{L}^{\epsilon}}) \right] \pm \mathbb{E} [g_L 1_{A_L}] - M_L
$$

$$
= \mathbb{E} \left[ g(X(T))1_{A_{L}^{\epsilon}} \right] + \mathbb{E} \left[ (g(X(T)) - g_L) 1_{A_L} \right] + \mathbb{E} [g_L 1_{A_L}] - M_L.
$$

Problem: Given $TOL > 0$, find the parameters for computing $M_L$ such that $|\mathcal{E}_L| < TOL$ with high probability, and with nearly optimal computational work.

Issues:

1. Simulate coupled pairs $(g_{\ell}, g_{\ell-1})$ for $\ell = 1, \ldots, L$.
2. Estimate accurately the global error components.
Theorem: Computational complexity of the Multilevel Hybrid Chernoff Tau-Leap is

\[
    w_L(TOL) = \left( \frac{C_A}{\theta} \sum_{\ell=0}^{L} \sqrt{\mathcal{V}_\ell \psi_\ell} \right)^2 TOL^{-2} = \mathcal{O}(TOL^{-2}).
\]

where \( \mathcal{V}_0 := \text{Var}[g_0 \mathbf{1}_{A_0}] \), \( \mathcal{V}_\ell := \text{Var}[g_\ell \mathbf{1}_{A_\ell} - g_{\ell-1} \mathbf{1}_{A_{\ell-1}}] \), \( \ell \geq 1 \) and \( \psi_\ell \) is the expected computational work per level of simulating coupled paths.
Numerical results: global error vs actual work

![Graph showing actual work vs. error bound for different models.]

Figure: Actual work for each one of the one hundred calibrations.
Part III

1. This is a new extension of our hybrid Multilevel Monte Carlo method.
2. The Reaction-splitting strategy is intended to deal with stiff problems.
3. A novel Control Variate based on a deterministic-time change approximation.
4. We preserve the computational complexity $O(TOL^{-2})$.
5. Some numerical results.
Example: Virus kinetics

- $E \xrightarrow{1} E + G$, the viral template (E) forms a viral genome (G).
- $G \xrightarrow{0.025} E$, the genome generates a new template.
- $E \xrightarrow{1000} E + S$, a viral structural protein (S) is generated.
- $G + S \xrightarrow{7.5 \times 10^{-6}} V$, the virus (V) is produced.
- $E \xrightarrow{0.25} \emptyset, S \xrightarrow{2} \emptyset$ degradation reactions.

\[
\nu := \begin{pmatrix}
1 & 0 & 0 & 0 \\
-1 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 \\
-1 & -1 & 0 & 1 \\
0 & 0 & -1 & 0 \\
0 & -1 & 0 & 0 \\
\end{pmatrix}^\text{tr}
\]

and

\[
a(X) := \begin{pmatrix}
E \\
0.025 \ G \\
1000 \ E \\
7.5 \times 10^{-6} \ G \ S \\
0.25 \ E \\
2 \ S \\
\end{pmatrix},
\]

where $X(t) = (G(t), S(t), E(t), V(t))$. 
Example: Virus kinetics

Estimate the expected number of produced viruses (V) at time $T=20$ (in days) up to a given $TOL$ with high confidence.

Figure: Exact paths of the time evolution of the species in the virus kinetics (VRK) example.
Figure: The quotient $\hat{W}_{Mix}/\hat{W}_{Hyb}$ starts at 2% and reaches its asymptotic value of 6%.
Final comments

• A novel mixed algorithm that allows us to approach problems in which there is a natural partition on the reaction channels.

• An 3-phase estimation procedure that provides the elements for the multilevel Monte Carlo setting (one-step exit probabilities, time meshes and number of mixed paths), which optimizes the computational work.

• The complexity of the method is of order $O(TOL^{-2})$, like in the SSA, but with a better constant!

• A control variate for the variance of the QoI when using single-level paths (applies for the hybrid and mixed methods).

• We achieved speedups of $10^2 – 10^4$ with the control variate and the mixed MLMC for large systems (200 reactions & species).


References IV


