Computational methods for the stochastic models of biochemical systems

David F. Anderson*

*anderson@math.wisc.edu

Department of Mathematics

University of Wisconsin - Madison

CWI: Stochastics and Nonlinear Dynamics in the Life Sciences

November 15th, 2012

Stochastic models of biochemical systems

Outline

1. Construct mathematical representation for most common stochastic model – continuous time Markov chains (master equation).

2. Discuss multi-level Monte Carlo: new (unbiased) method to approximate expectations.

- Start with some examples.

Example: ODE Lotka-Volterra predator-prey model

Think of A as a prey and B as a predator.

$$A \xrightarrow{\kappa_1} 2A, \qquad A + B \xrightarrow{\kappa_2} 2B, \qquad B \xrightarrow{\kappa_3} \emptyset,$$

with $\kappa_1 = 2$, $\kappa_2 = .002$, $\kappa_3 = 2$.

Example: ODE Lotka-Volterra predator-prey model

Think of A as a prey and B as a predator.

$$A \xrightarrow{\kappa_1} 2A, \qquad A + B \xrightarrow{\kappa_2} 2B, \qquad B \xrightarrow{\kappa_3} \emptyset,$$

with $\kappa_1 = 2$, $\kappa_2 = .002$, $\kappa_3 = 2$.

Deterministic model. Let $x(t) = [\#\text{ prey at } t, \#\text{ predator at } t]^T$

$$\dot{x(t)} = \kappa_1 x_1(t) \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \kappa_2 x_1(t) x_2(t) \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \kappa_3 x_2(t) \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

or

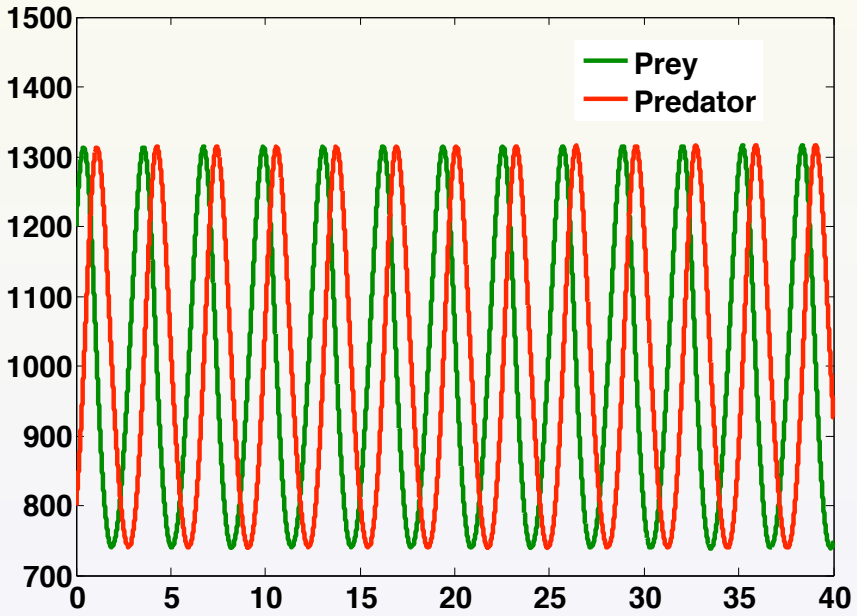
$$x(t) = x(0) + \kappa_1 \int_0^t x_1(s)ds \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \kappa_2 \int_0^t x_1(s)x_2(s)ds \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \kappa_3 \int_0^t x_2(s)ds \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

Lotka-Volterra

Think of A as a prey and B as a predator.

$$A \xrightarrow{\kappa_1} 2A, \qquad A + B \xrightarrow{\kappa_2} 2B, \qquad B \xrightarrow{\kappa_3} \emptyset,$$

with $\kappa_1 = 2$, $\kappa_2 = .002$, $\kappa_3 = 2$.



Biological example: transcription-translation

Gene transcription & translation:

$$G \xrightarrow{\kappa_1} G + M \quad \textit{transcription}$$

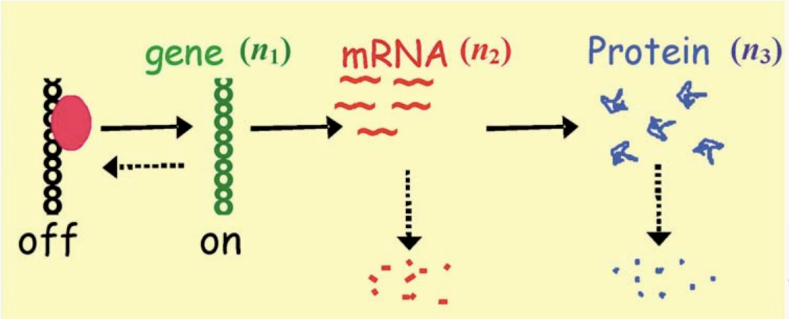
$$M \xrightarrow{\kappa_2} M + P \quad \textit{translation}$$

$$M \xrightarrow{\kappa_3} \emptyset \quad \textit{degradation}$$

$$P \xrightarrow{\kappa_4} \emptyset \quad \textit{degradation}$$

$$G + P \underset{\kappa_{-5}}{\overset{\kappa_5}{\rightleftarrows}} B \quad \text{Binding/unbinding of Gene}$$

Cartoon representation:



[1]

[1] J. Paulsson, Physics of Life Reviews, **2**, 2005 157 – 175.

Another example: Viral infection

Let

1. $T =$ viral template.
2. $G =$ viral genome.
3. $S =$ viral structure.
4. $V =$ virus.

Reactions:

R1) $T +$ "stuff" $\stackrel{\kappa_1}{\to} T + G$	$\kappa_1 = 1$
R2) $G \stackrel{\kappa_2}{\to} T$	$\kappa_2 = 0.025$
R3) $T +$ "stuff" $\stackrel{\kappa_3}{\to} T + S$	$\kappa_3 = 1000$
R4) $T \stackrel{\kappa_4}{\to} \emptyset$	$\kappa_4 = 0.25$
R5) $S \stackrel{\kappa_5}{\to} \emptyset$	$\kappa_5 = 2$
R6) $G + S \stackrel{\kappa_6}{\to} V$	$\kappa_6 = 7.5 \times 10^{-6}$

- R. Srivastava, L. You, J. Summers, and J. Yin, J. Theoret. Biol., 2002.
- E. Haseltine and J. Rawlings, J. Chem. Phys, 2002.
- K. Ball, T. Kurtz, L. Popovic, and G. Rempala, Annals of Applied Probability, 2006.
- W. E, D. Liu, and E. Vanden-Eijden, J. Comput. Phys, 2006.

Some examples

E. coli Heat Shock Response Model. 9 species, 18 reactions.

Reaction	Intensity	Reaction	Intensity
$\emptyset \to A_8$	4.00×10^{0}	$A_6 + A_8 \to A_9$	$3.62 \times 10^{-4} X_{A_6} X_{A_8}$
$A_2 \to A_3$	$7.00 \times 10^{-1} X_{A_2}$	$A_8 \to \emptyset$	$9.99 \times 10^{-5} X_{A_8}$
$A_3 \to A_2$	$1.30 \times 10^{-1} X_{A_3}$	$A_9 \to A_6 + A_8$	$4.40 \times 10^{-5} X_{A_9}$
$\emptyset \xrightarrow{A_1} A_2$	$7.00 \times 10^{-3} X_{A_1}$	$\emptyset \to A_1$	1.40×10^{-5}
$\text{stuff} + A_3 \to A_5 + A_2$	$6.30 \times 10^{-3} X_{A_3}$	$A_1 \to \emptyset$	$1.40 \times 10^{-6} X_{A_1}$
$\text{stuff} + A_3 \to A_4 + A_2$	$4.88 \times 10^{-3} X_{A_3}$	$A_7 \xrightarrow{A_4} A_6$	$1.42 \times 10^{-6} X_{A_4} X_{A_7}$
$\text{stuff} + A_3 \to A_6 + A_2$	$4.88 \times 10^{-3} X_{A_3}$	$A_5 \to \emptyset$	$1.80 \times 10^{-8} X_{A_5}$
$A_7 \to A_2 + A_6$	$4.40 \times 10^{-4} X_{A_7}$	$A_6 \to \emptyset$	$6.40 \times 10^{-10} X_{A_6}$
$A_2 + A_6 \to A_7$	$3.62 \times 10^{-4} X_{A_2} X_{A_6}$	$A_4 \to \emptyset$	$7.40 \times 10^{-11} X_{A_4}$

[2]

[2] Hye Won Kang, presentation at SPA in 2007.

Modeling

1. These models (and much more complicated ones) have historically been predominantly modeled using ODEs.

2. However,
   - there are often low numbers of molecules, the system jumps to new state by non-trivial amount: $1 \gg 0$.
   - Technology such as Green fluorescent protein showed randomness in outcome.

ODEs often seemed to be the wrong modeling choice.

Specifying infinitesimal behavior

Q: What is a better modeling choice? Should be

1. discrete space, since counting molecules, and
2. stochastic dynamics.

Let's return to development of ODEs.

An ordinary differential equation is specified by describing how a function should vary over a small period of time

$$X(t + \Delta t) - X(t) \approx F(X(t))\Delta t$$

Specifying infinitesimal behavior

Q: What is a better modeling choice? Should be

1. discrete space, since counting molecules, and
2. stochastic dynamics.

Let's return to development of ODEs.

An ordinary differential equation is specified by describing how a function should vary over a small period of time

$$X(t + \Delta t) - X(t) \approx F(X(t))\Delta t$$

A more precise description (consider a telescoping sum)

$$X(t) = X(0) + \int_0^t F(X(s))ds$$

Infinitesimal behavior for jump processes

We are interested in functions that are piecewise constant and random.

Changes, when they occur, won't be small. If "reaction k" occurs at time t,

$$X(t) - X(t-) = \zeta_k \in \mathbb{Z}^d$$

Infinitesimal behavior for jump processes

We are interested in functions that are piecewise constant and random.

Changes, when they occur, won't be small. If "reaction k" occurs at time t,

$$X(t) - X(t-) = \zeta_k \in \mathbb{Z}^d$$

What is small? The probability of seeing a jump of a particular size.

$$P\{X(t + \Delta t) - X(t) = \zeta_k \mid \mathcal{F}_t\} \approx \lambda_{\zeta_k}(t)\Delta t$$

Question: Can we specify the λ_{ζ_k} in some way that determines X?

- For the ODE, F depended on X.
- Maybe λ_{ζ_k} should depend on X?

Simple model

For example, consider the simple system

$$A + B \to C$$

where one molecule each of A and B is being converted to one of C.

Simple model

For example, consider the simple system

$$A + B \to C$$

where one molecule each of A and B is being converted to one of C.

Intuition for standard stochastic model:

$$P\{\text{reaction occurs in } (t, t+\Delta t] \big| \mathcal{F}_t\} \approx \kappa X_A(t) X_B(t) \Delta t$$

where

- κ is a positive constant, the reaction rate constant.
- $\mathcal{F}_t$ is all the information pertaining to the process up through time t.

Can we specify a reasonable model satisfying this assumption?

Answer: yes, but we need Poisson processes.

Background information: The Poisson process

Will view a Poisson process, $Y(\cdot)$, through the lens of an underlying point process.

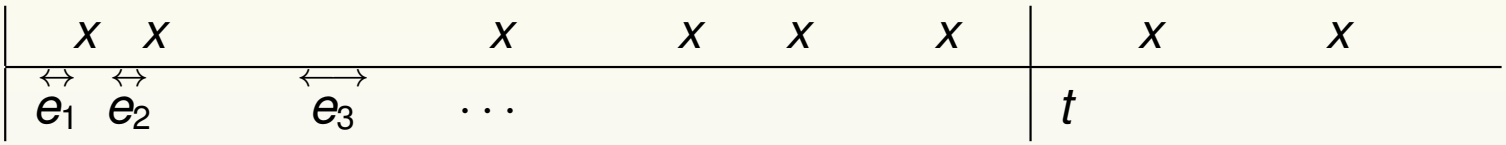
(a) Let $\{e_i\}$ be i.i.d. exponential random variables with parameter one.

Background information: The Poisson process

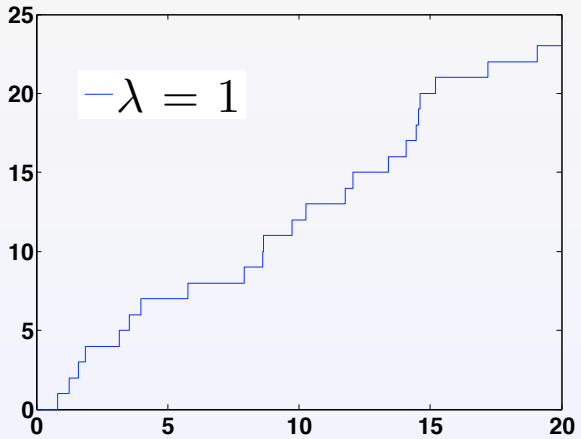
Will view a Poisson process, $Y(\cdot)$, through the lens of an underlying point process.

(a) Let $\{e_i\}$ be i.i.d. exponential random variables with parameter one.

(b) Now, put points down on a line with spacing equal to the e_i:



- Let $Y_1(t)$ denote the number of points hit by time t.
- In the figure above, $Y_1(t) = 6$.

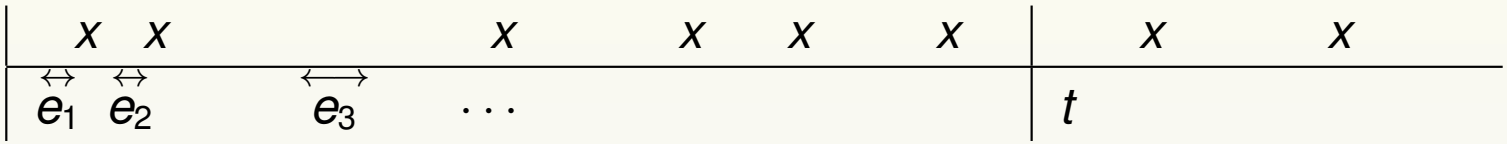


The Poisson process

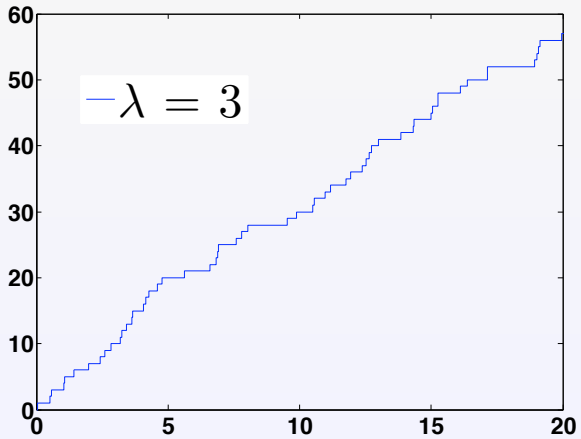
Let

- Y_1 be a unit rate Poisson process.
- Define $Y_\lambda(t) \equiv Y_1(\lambda t)$,

Then Y_λ is a Poisson process with parameter λ.



Intuition: The Poisson process with rate λ is simply the number of points hit (of the unit-rate point process) when we run along the time frame at rate λ.



The Poisson process

There is no reason λ needs to be constant in time, in which case

$$Y_\lambda(t) \equiv Y\left(\int_0^t \lambda(s)ds\right)$$

is a non-homogeneous Poisson process with propensity/intensity $\lambda(t) \geq 0$.

Thus

$$P\{Y_\lambda(t+\Delta t) - Y_\lambda(t) > 0 | \mathcal{F}_t\} = 1 - \exp\left\{-\int_t^{t+\Delta t} \lambda(s)ds\right\} \approx \lambda(t)\Delta t.$$

The Poisson process

There is no reason λ needs to be constant in time, in which case

$$Y_\lambda(t) \equiv Y\left(\int_0^t \lambda(s)ds\right)$$

is a non-homogeneous Poisson process with propensity/intensity $\lambda(t) \geq 0$.

Thus

$$P\{Y_\lambda(t+\Delta t) - Y_\lambda(t) > 0 | \mathcal{F}_t\} = 1 - \exp\left\{-\int_t^{t+\Delta t} \lambda(s)ds\right\} \approx \lambda(t)\Delta t.$$

Points:

1. We have "changed time" to convert a unit-rate Poisson process to one which has rate or intensity or propensity $\lambda(t)$.

2. Will use similar time changes of unit-rate processes to build models of interest.

Return to models of interest

Consider the simple system

$$A + B \to C$$

where one molecule each of A and B is being converted to one of C.

Intuition for standard stochastic model:

$$P\{\text{reaction occurs in } (t, t+\Delta t] \,\big|\, \mathcal{F}_t\} \approx \kappa X_A(t) X_B(t) \Delta t$$

where

- κ is a positive constant, the reaction rate constant.
- $\mathcal{F}_t$ is all the information pertaining to the process up through time t.

Models of interest

$$A + B \to C$$

Simple book-keeping says: if

$$X(t) = \begin{bmatrix} X_A(t) \\ X_B(t) \\ X_C(t) \end{bmatrix}$$

gives the state at time t, then

$$X(t) = X(0) + R(t) \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix},$$

where

- $R(t)$ is the # of times the reaction has occurred by time t and
- $X(0)$ is the initial condition.

Goal: represent $R(t)$ in terms of Poisson process.

Models of interest

Recall that for $A + B \to C$ our intuition was to specify infinitesimal behavior

$$P\{\text{reaction occurs in } (t, t + \Delta t] \big| \mathcal{F}_t\} \approx \kappa X_A(t) X_B(t) \Delta t,$$

Models of interest

Recall that for $A + B \to C$ our intuition was to specify infinitesimal behavior

$$P\{\text{reaction occurs in } (t, t+\Delta t] \big| \mathcal{F}_t\} \approx \kappa X_A(t) X_B(t) \Delta t,$$

and that for a counting process with specified intensity $\lambda(t)$ we have

$$P\{ Y_\lambda(t+\Delta t) - Y_\lambda(t) = 1 | \mathcal{F}_t\} \approx \lambda(t) \Delta t.$$

Models of interest

Recall that for $A + B \to C$ our intuition was to specify infinitesimal behavior

$$P\{\text{reaction occurs in } (t, t+\Delta t] \big| \mathcal{F}_t\} \approx \kappa X_A(t) X_B(t) \Delta t,$$

and that for a counting process with specified intensity $\lambda(t)$ we have

$$P\{Y_\lambda(t+\Delta t) - Y_\lambda(t) = 1 | \mathcal{F}_t\} \approx \lambda(t) \Delta t.$$

This suggests we can model

$$R(t) = Y\left(\int_0^t \kappa X_A(s) X_B(s) ds\right)$$

where Y is a unit-rate Poisson process.

Models of interest

Recall that for $A + B \to C$ our intuition was to specify infinitesimal behavior

$$P\{\text{reaction occurs in } (t, t+\Delta t] \big| \mathcal{F}_t\} \approx \kappa X_A(t) X_B(t) \Delta t,$$

and that for a counting process with specified intensity $\lambda(t)$ we have

$$P\{Y_\lambda(t+\Delta t) - Y_\lambda(t) = 1 | \mathcal{F}_t\} \approx \lambda(t) \Delta t.$$

This suggests we can model

$$R(t) = Y\left(\int_0^t \kappa X_A(s) X_B(s) ds\right)$$

where Y is a unit-rate Poisson process.
Hence

$$\begin{pmatrix} X_A(t) \\ X_B(t) \\ X_C(t) \end{pmatrix} \equiv X(t) = X(0) + \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix} Y\left(\int_0^t \kappa X_A(s) X_B(s) ds\right).$$

This equation uniquely determines X for all $t \geq 0$.

Build up model: Random time change representation of Kurtz

- Now consider a network of reactions involving d chemical species, $S_1, \ldots, S_d$:

$$\sum_{i=1}^{d} \nu_{ik} S_i \longrightarrow \sum_{i=1}^{d} \nu'_{ik} S_i$$

Denote reaction vector as

$$\zeta_k = \nu'_k - \nu_k,$$

so that if reaction k occurs at time t

$$X(t) = X(t-) + \zeta_k.$$

Build up model: Random time change representation of Kurtz

- Now consider a network of reactions involving d chemical species, $S_1, \ldots, S_d$:

$$\sum_{i=1}^{d} \nu_{ik} S_i \longrightarrow \sum_{i=1}^{d} \nu'_{ik} S_i$$

Denote reaction vector as

$$\zeta_k = \nu'_k - \nu_k,$$

so that if reaction k occurs at time t

$$X(t) = X(t-) + \zeta_k.$$

- The intensity (or propensity) of kth reaction is $\lambda_k : \mathbb{Z}^d_{\geq 0} \to \mathbb{R}$.
- By analogy with before:

$$X(t) = X(0) + \sum_k R_k(t)\zeta_k,$$

with

$$X(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k(X(s))ds \right) \zeta_k,$$

Y_k are independent, unit-rate Poisson processes.

Mass-action kinetics

The standard intensity function chosen is mass-action kinetics:

$$\lambda_k(x) = \kappa_k(\prod_i \nu_{ik}!)\binom{x}{\nu_k} = \kappa_k \prod_i \frac{x_i!}{(x_i - \nu_{ik})!}.$$

Example: If $S_1 \to$ anything, then $\lambda_k(x) = \kappa_k x_1$.

Example: If $S_1 + S_2 \to$ anything, then $\lambda_k(x) = \kappa_k x_1 x_2$.

Example: If $2S_2 \to$ anything, then $\lambda_k(x) = \kappa_k x_2(x_2 - 1)$.

Other ways to understand model

The infinitesimal generator of a Markov process determines the process:

$$\mathcal{A}f(x) \stackrel{def}{=} \lim_{h \to 0} \frac{1}{h} \left[\mathbb{E}_x f(X(h)) - f(x) \right]$$

Other ways to understand model

The infinitesimal generator of a Markov process determines the process:

$$\begin{aligned} \mathcal{A}f(x) &\stackrel{def}{=} \lim_{h\to 0} \frac{1}{h}\left[\mathbb{E}_x f(X(h)) - f(x)\right] \\ &= \lim_{h\to 0} \frac{1}{h}\left[\sum_k (f(x+\zeta_k) - f(x))P(R_k(h)=1)\right] + O(h) \end{aligned}$$

Other ways to understand model

The infinitesimal generator of a Markov process determines the process:

$$\begin{aligned}
\mathcal{A}f(x) &\stackrel{def}{=} \lim_{h\to 0} \frac{1}{h}\left[\mathbb{E}_x f(X(h)) - f(x)\right] \\
&= \lim_{h\to 0} \frac{1}{h}\left[\sum_k (f(x+\zeta_k) - f(x)) P(R_k(h) = 1)\right] + O(h) \\
&= \lim_{h\to 0} \frac{1}{h}\left[\sum_k (f(x+\zeta_k) - f(x))\lambda_k(x)h\right] + O(h)
\end{aligned}$$

Other ways to understand model

The infinitesimal generator of a Markov process determines the process:

$$\begin{aligned}
\mathcal{A}f(x) &\stackrel{def}{=} \lim_{h\to 0}\frac{1}{h}\left[\mathbb{E}_x f(X(h)) - f(x)\right] \\
&= \lim_{h\to 0}\frac{1}{h}\left[\sum_k (f(x+\zeta_k) - f(x))P(R_k(h)=1)\right] + O(h) \\
&= \lim_{h\to 0}\frac{1}{h}\left[\sum_k (f(x+\zeta_k) - f(x))\lambda_k(x)h\right] + O(h) \\
&= \sum_k \lambda_k(x)(f(x+\zeta_k) - f(x)).
\end{aligned}$$

Other ways to understand model

And we have Dynkin's formula (See Ethier and Kurtz, 1986, Ch. 1)

$$\mathbb{E}f(X(t)) - f(X_0) = \mathbb{E}\int_0^t \mathcal{A}f(X(s))ds,$$

Other ways to understand model

And we have Dynkin's formula (See Ethier and Kurtz, 1986, Ch. 1)

$$\mathbb{E}f(X(t)) - f(X_0) = \mathbb{E}\int_0^t \mathcal{A}f(X(s))ds,$$

Letting $f(y) = 1_x(y)$, above so that

$$\mathbb{E}[f(X(t))] = P\{X(t) = x\} = p_x(t),$$

gives Kolmogorov forward equation (chemical master equation)

$$p_t'(x) = \sum_k \lambda(x - \zeta_k)p_t(x - \zeta_k) - p_t(x)\sum_k \lambda_k(x)$$

Equivalence of formulations

We now have three ways of making the infinitesimal specification

$$P\{X(t+\Delta t) - X(t) = \zeta_k \mid \mathcal{F}_t^X\} \approx \lambda_k(X(t))\Delta t$$

precise:

1. The stochastic equation:

$$X(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k(X(s))ds \right) \zeta_k$$

2. The process is Markov with infinitesimal generator

$$(\mathcal{A}f)(x) = \sum_k \lambda_k(x)(f(x+\zeta_k) - f(x))$$

3. The master (forward) equation for the probability distributions:

$$p_x'(t) = \sum_k \lambda_k(x-\zeta_k)p_t(x-\zeta_k) - p_t(x)\sum_k \lambda_k(x)$$

Fortunately, if the solution of the stochastic equation doesn't blow up, the three are equivalent.

This model is an example of a continuous time Markov chain.

Example: ODE Lotka-Volterra predator-prey model

Think of A as a prey and B as a predator.

$$A \xrightarrow{\kappa_1} 2A, \qquad A + B \xrightarrow{\kappa_2} 2B, \qquad B \xrightarrow{\kappa_3} \emptyset,$$

with $\kappa_1 = 2$, $\kappa_2 = .002$, $\kappa_3 = 2$.

Deterministic model. Let $x(t) = [\#\text{prey}, \#\text{predators}]^T$

$$x(t) = x(0) + \kappa_1 \int_0^t x_1(s)ds \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \kappa_2 \int_0^t x_1(s)x_2(s)ds \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \kappa_3 \int_0^t x_2(s)ds \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

Example: ODE Lotka-Volterra predator-prey model

Think of A as a prey and B as a predator.

$$A \xrightarrow{\kappa_1} 2A, \qquad A + B \xrightarrow{\kappa_2} 2B, \qquad B \xrightarrow{\kappa_3} \emptyset,$$

with $\kappa_1 = 2$, $\kappa_2 = .002$, $\kappa_3 = 2$.

Deterministic model. Let $x(t) = [\#\text{prey}, \#\text{predators}]^T$

$$x(t) = x(0) + \kappa_1 \int_0^t x_1(s)ds \begin{bmatrix} 1 \\ 0 \end{bmatrix} + \kappa_2 \int_0^t x_1(s)x_2(s)ds \begin{bmatrix} -1 \\ 1 \end{bmatrix} + \kappa_3 \int_0^t x_2(s)ds \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

Stochastic model. Let $X(t) = [\#\text{prey}, \#\text{predators}]^T$

$$X(t) = X(0) + Y_1\left(\kappa_1 \int_0^t X_1(s)ds\right) \begin{bmatrix} 1 \\ 0 \end{bmatrix} + Y_2\left(\kappa_2 \int_0^t X_1(s)X_2(s)ds\right) \begin{bmatrix} -1 \\ 1 \end{bmatrix}$$

$$+ Y_3\left(\kappa_3 \int_0^t X_2(s)ds\right) \begin{bmatrix} 0 \\ -1 \end{bmatrix}$$

Computation of expectations

Problem for today: Approximate $\mathbb{E}f(X(T))$ to some desired tolerance, $\epsilon > 0$.

Computation of expectations

Problem for today: Approximate $\mathbb{E}f(X(T))$ to some desired tolerance, $\epsilon > 0$.

Easy!:

- Simulate the CTMC exactly,
- generate independent paths, $X_{[i]}(t)$, use the unbiased/consistent estimator

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(X_{[i]}(t)) \overset{a.s.}{\to} \mathbb{E}f(X(t)).$$

- stop when desired confidence interval is $\pm\epsilon$.

What is the computational cost?

Recall,

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(X_{[i]}(t)).$$

Thus,

$$\mathrm{Var}(\mu_n) = \frac{1}{n}\mathrm{Var}(f(X(t))).$$

What is the computational cost?

Recall,

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(X_{[i]}(t)).$$

Thus,

$$\mathrm{Var}(\mu_n) = \frac{1}{n}\mathrm{Var}(f(X(t))).$$

So, if we want

$$\widehat{\sigma}_n = O(\epsilon),$$

we need

$$\frac{\sigma_{f(X(t))}}{\sqrt{n}} = O(\epsilon) \implies n = O\left(\sigma_{f(X(t))}\epsilon^{-2}\right).$$

What is the computational cost?

Recall,

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(X_{[i]}(t)).$$

Thus,

$$\mathrm{Var}(\mu_n) = \frac{1}{n}\mathrm{Var}(f(X(t))).$$

So, if we want

$$\widehat{\sigma}_n = O(\epsilon),$$

we need

$$\frac{\sigma_{f(X(t))}}{\sqrt{n}} = O(\epsilon) \implies n = O\left(\sigma_{f(X(t))}\epsilon^{-2}\right).$$

If $\overline{N}$ gives average cost (steps) of a path using exact algorithm:

$$\begin{aligned}\text{Total computational complexity} &= (\text{cost per path}) \times (\#\text{ paths}) \\ &= O(\overline{N}\sigma_{f(X(t))}\epsilon^{-2}).\end{aligned}$$

Can be bad if (i) $\overline{N} \cdot \sigma_{f(X(t))}$, and/or (ii) ϵ is small.

Benefits/drawbacks

Benefits:

1. Easy to implement.
2. Estimator

$$\mu_n = \frac{1}{n} \sum_{i=1}^{n} f(X_{[i]}(t))$$

is unbiased and consistent.

Benefits/drawbacks

Benefits:

1. Easy to implement.
2. Estimator

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(X_{[i]}(t))$$

   is unbiased and consistent.

Drawbacks:

1. The cost of $O(\overline{N}\sigma_{f(X(t))}\epsilon^{-2})$ could be prohibitively large.
2. For our models, we often have that $\overline{N}$ is very large.

Let's try an approximate method.

Tau-leaping: Euler's method

Explicit tau-leaping [3] or Euler's method, was first formulated by Dan Gillespie in this setting.

Tau-leaping is essentially an Euler approximation of $\int_0^t \lambda_k(X(s))ds$:

$$
\begin{aligned}
Z(h) &= Z(0) + \sum_k Y_k \left(\int_0^h \lambda_k(Z(s))\, ds \right) \zeta_k \\
&\approx Z(0) + \sum_k Y_k \Big(\lambda_k(Z(0))\, h \Big) \zeta_k \\
&\stackrel{d}{=} Z(0) + \sum_k \text{Poisson} \Big(\lambda_k(Z(0))\, h \Big) \zeta_k .
\end{aligned}
$$

[3] D. T. Gillespie, J. Chem. Phys., **115**, 1716 – 1733.

Euler's method

Path-wise representation for $Z(t)$ generated by Euler's method is

$$Z(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k(Z \circ \eta(s)) ds \right) \zeta_k,$$

where

$$\eta(s) = \left\lfloor \frac{s}{h} \right\rfloor h, \quad \Longrightarrow \eta(s) = t_n \quad \text{if} \quad t_n \leq s < t_{n+1} = t_n + h$$

is a step function giving left endpoints of time discretization.

Return to approximating $\mathbb{E}f(X(T))$

Let Z_ℓ denote an approximate processes generated with time discretization step of h_ℓ . Let

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(Z_{\ell,[i]}(t)).$$

Return to approximating $\mathbb{E}f(X(T))$

Let Z_ℓ denote an approximate processes generated with time discretization step of h_ℓ . Let

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(Z_{\ell,[i]}(t)).$$

We note

$$\begin{aligned}\mathbb{E}f(X(t)) - \mu_n &= \left[\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t))\right] + \mathbb{E}f(Z_L(t)) - \mu_n \\ &= \qquad \text{bias} \qquad + \qquad \text{statistical error}.\end{aligned}$$

Return to approximating $\mathbb{E}f(X(T))$

Let Z_ℓ denote an approximate processes generated with time discretization step of h_ℓ . Let

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(Z_{\ell,[i]}(t)).$$

We note

$$\begin{aligned}\mathbb{E}f(X(t)) - \mu_n &= \left[\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t))\right] + \mathbb{E}f(Z_L(t)) - \mu_n \\ &= \qquad \text{bias} \qquad + \qquad \text{statistical error}.\end{aligned}$$

Suppose have an order one method

$$\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t)) = O(h_\ell).$$

Return to approximating $\mathbb{E}f(X(T))$

Let Z_ℓ denote an approximate processes generated with time discretization step of h_ℓ . Let

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(Z_{\ell,[i]}(t)).$$

We note

$$\begin{aligned}\mathbb{E}f(X(t)) - \mu_n &= \left[\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t))\right] + \mathbb{E}f(Z_L(t)) - \mu_n \\ &= \text{bias} + \text{statistical error}.\end{aligned}$$

Suppose have an order one method

$$\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t)) = O(h_\ell).$$

We need:

1. $h_L = O(\epsilon)$.
2. $n = \epsilon^{-2}$.

Return to approximating $\mathbb{E}f(X(T))$

Let Z_ℓ denote an approximate processes generated with time discretization step of h_ℓ . Let

$$\mu_n = \frac{1}{n}\sum_{i=1}^{n} f(Z_{\ell,[i]}(t)).$$

We note

$$\begin{aligned}\mathbb{E}f(X(t)) - \mu_n &= \left[\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t))\right] + \mathbb{E}f(Z_L(t)) - \mu_n \\ &= \qquad \text{bias} \qquad + \qquad \text{statistical error}.\end{aligned}$$

Suppose have an order one method

$$\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t)) = O(h_\ell).$$

We need:

1. $h_L = O(\epsilon)$.
2. $n = \epsilon^{-2}$.

Suppose a path costs $O(\epsilon^{-1})$ steps. Then

$$\begin{aligned}\text{Total computational complexity} &= (\#\text{ paths}) \times (\text{cost per path}) \\ &= O(\epsilon^{-3}).\end{aligned}$$

Benefits/drawbacks

Benefits:

1. Can drastically lower the computational complexity of a problem if $\epsilon^{-1} \ll \overline{N}$.

$$\text{CC of using exact} = \overline{N}\epsilon^{-2}$$

$$\text{CC of using approximate} = \epsilon^{-1}\epsilon^{-2}.$$

Benefits/drawbacks

Benefits:

1. Can drastically lower the computational complexity of a problem if $\epsilon^{-1} \ll \overline{N}$.

$$\text{CC of using exact} = \overline{N}\epsilon^{-2}$$

$$\text{CC of using approximate} = \epsilon^{-1}\epsilon^{-2}.$$

Drawbacks:

1. Convergence results usually give order of convergence. Can't give a precise h_L. Bias is a problem.

2. Tau-leaping has problems: what happens if you go negative?

3. Gone away from an unbiased estimator.

Recap

- Exact simulation is unbiased/consistent but potentially unusably slow.
- Tau-leaping/Euler's method is fast (as fast as you want), but has a difficult to quantify bias.

- Wouldn't it be great to have a method with best of both?
  - Unbiased
  - As fast as tau-leaping with crude stepsize.

Multi-level Monte Carlo: Main idea (Mike Giles)

First recall notation:

$$X(t) \sim \text{exact process}, \qquad Z_\ell \sim \text{tau-leaping with step size of } h_\ell = \frac{1}{M^\ell}.$$

So, we can think of multiple version of tau-leaping:

$$Z_L, \; Z_{L-1}, \; Z_{L-1}, \; \ldots, \; Z_{\ell_0}.$$

Multi-level Monte Carlo: Main idea (Mike Giles)

First recall notation:

$$X(t) \sim \text{exact process}, \qquad Z_\ell \sim \text{tau-leaping with step size of } h_\ell = \frac{1}{M^\ell}.$$

So, we can think of multiple version of tau-leaping:

$$Z_L, \; Z_{L-1}, \; Z_{L-1}, \; \ldots, \; Z_{\ell_0}.$$

Now simply note:

$$\mathbb{E}f(X(t)) \quad = \quad \mathbb{E}[f(X(t)) - f(Z_L(t))] + \mathbb{E}f(Z_L(t))$$

Multi-level Monte Carlo: Main idea (Mike Giles)

First recall notation:

$$X(t) \sim \text{exact process}, \qquad Z_\ell \sim \text{tau-leaping with step size of } h_\ell = \frac{1}{M^\ell}.$$

So, we can think of multiple version of tau-leaping:

$$Z_L, \; Z_{L-1}, \; Z_{L-1}, \; \ldots, \; Z_{\ell_0}.$$

Now simply note:

$$\begin{aligned} \mathbb{E}f(X(t)) &= \mathbb{E}[f(X(t)) - f(Z_L(t))] + \mathbb{E}f(Z_L(t)) \\ &= \mathbb{E}[f(X(t)) - f(Z_L(t))] + \mathbb{E}[f(Z_L(t)) - f(Z_{L-1}(t))] + \mathbb{E}f(Z_{L-1}(t)) \end{aligned}$$

Multi-level Monte Carlo: Main idea (Mike Giles)

First recall notation:

$$X(t) \sim \text{exact process}, \qquad Z_\ell \sim \text{tau-leaping with step size of } h_\ell = \frac{1}{M^\ell}.$$

So, we can think of multiple version of tau-leaping:

$$Z_L, \; Z_{L-1}, \; Z_{L-1}, \; \ldots, \; Z_{\ell_0}.$$

Now simply note:

$$\begin{aligned}
\mathbb{E}f(X(t)) &= \mathbb{E}[f(X(t)) - f(Z_L(t))] + \mathbb{E}f(Z_L(t)) \\
&= \mathbb{E}[f(X(t)) - f(Z_L(t))] + \mathbb{E}[f(Z_L(t)) - f(Z_{L-1}(t))] + \mathbb{E}f(Z_{L-1}(t)) \\
&\vdots \\
&= \mathbb{E}[f(X(t)) - f(Z_L(t))] + \sum_{\ell=\ell_0+1}^{L} \mathbb{E}[f(Z_\ell(t)) - f(Z_{\ell-1}(t))] + \mathbb{E}f(Z_{\ell_0}(t)).
\end{aligned}$$

Multi-level Monte Carlo: Main idea (Mike Giles)

$$\mathbb{E}f(X(t)) \quad = \quad \mathbb{E}[f(X(t)) - f(Z_L(t))] + \sum_{\ell=\ell_0+1}^{L} \mathbb{E}[f(Z_\ell(t)) - f(Z_{\ell-1}(t))] + \mathbb{E}f(Z_{\ell_0}(t)).$$

For appropriate choices of n_0, n_ℓ, and n_E, we define the estimators for the three terms above via

$$\widehat{Q}_E \quad \stackrel{\text{def}}{=} \quad \frac{1}{n_E} \sum_{i=1}^{n_E} (f(X_{[i]}(T) - f(Z_{L,[i]}(T))),$$

$$\widehat{Q}_\ell \quad \stackrel{\text{def}}{=} \quad \frac{1}{n_\ell} \sum_{i=1}^{n_\ell} (f(Z_{\ell,[i]}(T)) - f(Z_{\ell-1,[i]}(T))), \quad \text{for } \ell \in \{\ell_0+1, \ldots, L\}$$

$$\widehat{Q}_0 \quad \stackrel{\text{def}}{=} \quad \frac{1}{n_0} \sum_{i=1}^{n_0} f(Z_{\ell_0,[i]}(T)),$$

and note that

$$\widehat{Q} \stackrel{\text{def}}{=} \widehat{Q}_E + \sum_{\ell=\ell_0}^{L} \widehat{Q}_\ell$$

is an unbiased/consistent estimator for $\mathbb{E}f(X(T))$.

So what is coupling and the variance?

Easier question: How do we generate non-homogeneous Poisson processes simultaneously

Suppose we want

1. non-homogeneous Poisson process with intensity $f(t)$ and
2. non-homogeneous Poisson process with intensity $g(t)$.

Easier question: How do we generate non-homogeneous Poisson processes simultaneously

Suppose we want

1. non-homogeneous Poisson process with intensity $f(t)$ and
2. non-homogeneous Poisson process with intensity $g(t)$.

We can can let Y_1, Y_2, and Y_3 be independent, unit-rate Poisson processes and define

$$Z_f(t) = Y_1\left(\int_0^t f(s) \wedge g(s) ds\right) + Y_2\left(\int_0^t f(s) - (f(s) \wedge g(s))\, ds\right),$$

$$Z_g(t) = Y_1\left(\int_0^t f(s) \wedge g(s) ds\right) + Y_3\left(\int_0^t g(s) - (f(s) \wedge g(s))\, ds\right),$$

Easier question: How do we generate non-homogeneous Poisson processes simultaneously

Suppose we want

1. non-homogeneous Poisson process with intensity $f(t)$ and
2. non-homogeneous Poisson process with intensity $g(t)$.

We can can let Y_1, Y_2, and Y_3 be independent, unit-rate Poisson processes and define

$$Z_f(t) = Y_1\left(\int_0^t f(s) \wedge g(s) ds\right) + Y_2\left(\int_0^t f(s) - (f(s) \wedge g(s))\, ds\right),$$
$$Z_g(t) = Y_1\left(\int_0^t f(s) \wedge g(s) ds\right) + Y_3\left(\int_0^t g(s) - (f(s) \wedge g(s))\, ds\right),$$

where we are using that, for example,

$$Y_1\left(\int_0^t f(s) \wedge g(s) ds\right) + Y_2\left(\int_0^t f(s) - (f(s) \wedge g(s))\, ds\right) \stackrel{d}{=} Y\left(\int_0^t f(s) ds\right),$$

where Y is a unit rate Poisson process.

Back to our processes

$$X(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k(X(s))ds \right) \zeta_k,$$

$$Z(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k(Z \circ \eta(s))ds \right) \zeta_k.$$

Back to our processes

$$X(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k(X(s))ds \right) \zeta_k,$$

$$Z(t) = X(0) + \sum_k Y_k \left(\int_0^t \lambda_k(Z \circ \eta(s))ds \right) \zeta_k.$$

Now couple

$$\begin{aligned} X(t) =& X(0) + \sum_k Y_{k,1} \left(\int_0^t \lambda_k(X(s)) \wedge \lambda_k(Z_\ell \circ \eta_\ell(s))ds \right) \zeta_k \\ &+ \sum_k Y_{k,2} \left(\int_0^t \lambda_k(X(s)) - \lambda_k(X(s)) \wedge \lambda_k(Z_\ell \circ \eta_\ell(s))ds \right) \zeta_k \end{aligned}$$

$$\begin{aligned} Z_\ell(t) =& Z_\ell(0) + \sum_k Y_{k,1} \left(\int_0^t \lambda_k(X(s)) \wedge \lambda_k(Z_\ell \circ \eta_\ell(s))ds \right) \zeta_k \\ &+ \sum_k Y_{k,3} \left(\int_0^t \lambda_k(Z_\ell \circ \eta_\ell(s)) - \lambda_k(X(s)) \wedge \lambda_k(Z_\ell \circ \eta_\ell(s))ds \right) \zeta_k \end{aligned}$$

For approximate processes

$$Z_\ell(t) = Z_\ell(0) + \sum_k Y_{k,1}\left(\int_0^t \lambda_k(Z_\ell \circ \eta_\ell(s)) \wedge \lambda_k(Z_{\ell-1} \circ \eta_{\ell-1}(s))ds\right)\zeta_k$$
$$+ \sum_k Y_{k,2}\left(\int_0^t \lambda_k(Z_\ell \circ \eta_\ell(s)) - \lambda_k(Z_\ell \circ \eta_\ell(s)) \wedge \lambda_k(Z_{\ell-1} \circ \eta_{\ell-1}(s))ds\right)\zeta_k$$

$$Z_{\ell-1}(t) = Z_{\ell-1}(0) + \sum_k Y_{k,1}\left(\int_0^t \lambda_k(Z_\ell \circ \eta_\ell(s)) \wedge \lambda_k(Z_{\ell-1} \circ \eta_{\ell-1}(s))ds\right)\zeta_k$$
$$+ \sum_k Y_{k,3}\left(\int_0^t \lambda_k(Z_{\ell-1} \circ \eta_{\ell-1}(s)) - \lambda_k(Z_\ell \circ \eta_\ell(s)) \wedge \lambda_k(Z_{\ell-1} \circ \eta_{\ell-1}(s))ds\right)\zeta_k,$$

Multi-level Monte Carlo: chemical kinetic setting

Can prove:

Theorem (A., Higham 2011)

Suppose (X, Z_ℓ) *satisfy coupling. Then, there exist positive constants* $C_1(T), C_2(T) > 0$*, such that*

$$\sup_{t \leq T} \mathbb{E}|X(t) - Z_\ell(t)|^2 \leq C_1(T) N^{-\rho} h_\ell + C_2(T) h_\ell^2.$$

[3] David F. Anderson and Desmond J. Higham, *Multi-level Monte Carlo for continuous time Markov chains, with applications in biochemical kinetics*, SIAM: Multiscale Modeling and Simulation, Vol. 10, No. 1, 146 - 179, 2012.

Multi-level Monte Carlo: chemical kinetic setting

Can prove:

Theorem (A., Higham 2011)

Suppose (X, Z_ℓ) *satisfy coupling. Then, there exist positive constants* $C_1(T), C_2(T) > 0$*, such that*

$$\sup_{t \leq T} \mathbb{E}|X(t) - Z_\ell(t)|^2 \leq C_1(T) N^{-\rho} h_\ell + C_2(T) h_\ell^2.$$

Theorem (A., Higham 2011)

Suppose $(Z_\ell, Z_{\ell-1})$ *satisfy coupling. Then, there exist positive constants* $C_1(T), C_2(T) > 0$*, such that*

$$\sup_{t \leq T} \mathbb{E}|Z_\ell(t) - Z_{\ell-1}(t)|^2 \leq C_1(T) N^{-\rho} h_\ell + C_2(T) h_\ell^2.$$

---

[3] David F. Anderson and Desmond J. Higham, *Multi-level Monte Carlo for continuous time Markov chains, with applications in biochemical kinetics*, SIAM: Multiscale Modeling and Simulation, Vol. 10, No. 1, 146 - 179, 2012.

Multi-level Monte Carlo: an unbiased estimator

For well chosen n_0, n_ℓ, and n_E. We have

$$\mathrm{Var}(\widehat{Q}) = \mathrm{Var}\left(\widehat{Q}_E + \sum_{\ell=\ell_0}^{L} \widehat{Q}_\ell\right) = O(\epsilon^2),$$

with

$$\text{Comp. cost} = \left[\epsilon^{-2}(N^{-\rho}h_L + h_L^2)\right]\overline{N} + \epsilon^{-2}\left(h_{\ell_0}^{-1} + \ln(\epsilon)^2 N^{-\rho} + \ln(\epsilon^{-1})\frac{1}{M-1}\right)$$

Multi-level Monte Carlo: an unbiased estimator

Some observations:

1. Weak error plays no role in analysis: free to choose h_L.

Multi-level Monte Carlo: an unbiased estimator

Some observations:

1. Weak error plays no role in analysis: free to choose h_L.

2. Common problems associated with tau-leaping

   - Negativity of species numbers,

   does not matter. Just define process in a sensible way.

Multi-level Monte Carlo: an unbiased estimator

Some observations:

1. Weak error plays no role in analysis: free to choose h_L.

2. Common problems associated with tau-leaping

   - Negativity of species numbers,

   does not matter. Just define process in a sensible way.

3. The method is unbiased/consistent.

Example

Consider a model of gene transcription and translation:

$$G \xrightarrow{25} G + M,$$

$$M \xrightarrow{1000} M + P,$$

$$P + P \xrightarrow{0.001} D,$$

$$M \xrightarrow{0.1} \emptyset,$$

$$P \xrightarrow{1} \emptyset.$$

Suppose:

1. initialize with: $G = 1$, $M = 0$, $P = 0$, $D = 0$,
2. want to estimate the expected number of dimers at time $T = 1$,
3. to an accuracy of $\pm$ 1.0 with 95% confidence.

Example

Method: Exact algorithm with crude Monte Carlo.

Approximation	# paths	CPU Time	# updates
3,714.2 $\pm$ 1.0	4,740,000	149,000 CPU S (41 hours!)	8.27 $\times 10^{10}$

Example

Method: Exact algorithm with crude Monte Carlo.

Approximation	# paths	CPU Time	# updates
3,714.2 $\pm$ 1.0	4,740,000	149,000 CPU S (41 hours!)	8.27 $\times 10^{10}$

Method: Euler tau-leaping with crude Monte Carlo.

Step-size	Approximation	# paths	CPU Time	# updates
$h = 3^{-7}$	3,712.3 $\pm$ 1.0	4,750,000	13,374.6 S	6.2×10^{10}
$h = 3^{-6}$	3,707.5 $\pm$ 1.0	4,750,000	6,207.9 S	2.1×10^{10}
$h = 3^{-5}$	3,693.4 $\pm$ 1.0	4,700,000	2,803.9 S	6.9×10^{9}
$h = 3^{-4}$	3,654.6 $\pm$ 1.0	4,650,000	1,219.0 S	2.6×10^{9}

Example

Method: Exact algorithm with crude Monte Carlo.

Approximation	# paths	CPU Time	# updates
3,714.2 $\pm$ 1.0	4,740,000	149,000 CPU S (41 hours!)	8.27 $\times 10^{10}$

Method: unbiased MLMC with $\ell_0 = 2$, and M and L detailed below.

Step-size parameters	Approx.	CPU Time	# updates
$M = 3$, $L = 6$	3,713.9 $\pm$ 1.0	1,063.3 S	1.1 $\times 10^9$
$M = 3$, $L = 5$	3,714.7 $\pm$ 1.0	1,114.9 S	9.4 $\times 10^8$
$M = 3$, $L = 4$	3,714.2 $\pm$ 1.0	1,656.6 S	1.0 $\times 10^9$
$M = 4$, $L = 4$	3714.2 $\pm$ 1.0	1,334.8 S	1.1 $\times 10^9$
$M = 4$, $L = 5$	3,713.8 $\pm$ 1.0	1,014.9 S	1.1 $\times 10^9$

- the exact algorithm with crude Monte Carlo demanded
  - 140 times more CPU time

  than our unbiased MLMC estimator.

Some conclusions about this method

1. Gillespie's algorithm is by far the most common way to compute expectations:
    1.1 Means.
    1.2 Variances.
    1.3 Probabilities.
2. The new method (MLMC) also performs this task with no bias (exact).

Some conclusions about this method

1. Gillespie's algorithm is by far the most common way to compute expectations:
    1.1 Means.
    1.2 Variances.
    1.3 Probabilities.
2. The new method (MLMC) also performs this task with no bias (exact).
3. Will often be many orders of magnitude faster than Gillespie.

Some conclusions about this method

1. Gillespie's algorithm is by far the most common way to compute expectations:
    1.1 Means.
    1.2 Variances.
    1.3 Probabilities.
2. The new method (MLMC) also performs this task with no bias (exact).
3. Will often be many orders of magnitude faster than Gillespie.
4. Harder to implement, but open source software will be developed.
5. Makes no use of any specific structure or scaling in the problem.

Another example: Viral infection

Let

1. $T =$ viral template.
2. $G =$ viral genome.
3. $S =$ viral structure.
4. $V =$ virus.

Reactions:

$$
\begin{array}{ll}
\text{R1) } T + \text{stuff} \xrightarrow{\kappa_1} T + G & \kappa_1 = 1 \\
\text{R2) } G \xrightarrow{\kappa_2} T & \kappa_2 = 0.025 \\
\text{R3) } T + \text{stuff} \xrightarrow{\kappa_3} T + S & \kappa_3 = 1000 \\
\text{R4) } T \xrightarrow{\kappa_4} \emptyset & \kappa_4 = 0.25 \\
\text{R5) } S \xrightarrow{\kappa_5} \emptyset & \kappa_5 = 2 \\
\text{R6) } G + S \xrightarrow{\kappa_6} V & \kappa_6 = 7.5 \times 10^{-6}
\end{array}
$$

- R. Srivastava, L. You, J. Summers, and J. Yin, J. Theoret. Biol., 2002.
- E. Haseltine and J. Rawlings, J. Chem. Phys, 2002.
- K. Ball, T. Kurtz, L. Popovic, and G. Rempala, Annals of Applied Probability, 2006.
- W. E, D. Liu, and E. Vanden-Eijden, J. Comput. Phys, 2006.

Another example: Viral infection

Stochastic equations for $X = (X_G, X_S, X_T, X_V)$ are

$$X_1(t) = X_1(0) + Y_1\left(\int_0^t X_3(s)ds\right) - Y_2\left(0.025\int_0^t X_1(s)ds\right) - Y_6\left(7.5\times 10^{-6}\int_0^t X_1(s)X_2(s)ds\right)$$

$$X_2(t) = X_2(0) + Y_3\left(1000\int_0^t X_3(s)ds\right) - Y_5\left(2\int_0^t X_2(s)ds\right) - Y_6\left(7.5\times 10^{-6}\int_0^t X_1(s)X_2(s)ds\right)$$

$$X_3(t) = X_3(0) + Y_2\left(0.025\int_0^t X_1(s)ds\right) - Y_4\left(0.25\int_0^t X_3(s)ds\right)$$

$$X_4(t) = X_4(0) + Y_6\left(7.5\times 10^{-6}\int_0^t X_1(s)X_2(s)ds\right).$$

Another example: Viral infection

Reactions:

R1) $T+$ stuff $\stackrel{\kappa_1}{\to} T+G$	$\kappa_1 = 1$
R2) $G \stackrel{\kappa_2}{\to} T$	$\kappa_2 = 0.025$
R3) $T+$ stuff $\stackrel{\kappa_3}{\to} T+S$	$\kappa_3 = 1000$
R4) $T \stackrel{\kappa_4}{\to} \emptyset$	$\kappa_4 = 0.25$
R5) $S \stackrel{\kappa_5}{\to} \emptyset$	$\kappa_5 = 2$
R6) $G+S \stackrel{\kappa_6}{\to} V$	$\kappa_6 = 7.5 \times 10^{-6}$

If $T > 0$,

- reactions 3 and 5 are much faster than others.
- Looks like S is approximately Poisson$(500 \times T)$.

Another example: Viral infection

Reactions:

R1) $T+$ stuff $\stackrel{\kappa_1}{\to} T+G$	$\kappa_1 = 1$
R2) $G \stackrel{\kappa_2}{\to} T$	$\kappa_2 = 0.025$
R3) $T+$ stuff $\stackrel{\kappa_3}{\to} T+S$	$\kappa_3 = 1000$
R4) $T \stackrel{\kappa_4}{\to} \emptyset$	$\kappa_4 = 0.25$
R5) $S \stackrel{\kappa_5}{\to} \emptyset$	$\kappa_5 = 2$
R6) $G+S \stackrel{\kappa_6}{\to} V$	$\kappa_6 = 7.5 \times 10^{-6}$

If $T > 0$,

- reactions 3 and 5 are much faster than others.
- Looks like S is approximately Poisson$(500 \times T)$.

Can average out to get approximate process $Z(t)$.

Another example: Viral infection

Approximate process satisfies.

$$\begin{aligned} Z_1(t) &= X_1(0) + Y_1\left(\int_0^t Z_3(s)ds\right) - Y_2\left(0.025\int_0^t Z_1(s)ds\right) \\ &\quad - Y_6\left(3.75\times 10^{-3}\int_0^t Z_1(s)Z_3(s)ds\right) \\ Z_3(t) &= X_3(0) + Y_2\left(0.025\int_0^t Z_1(s)ds\right) - Y_4\left(0.25\int_0^t Z_3(s)ds\right) \\ Z_4(t) &= X_4(0) + Y_6\left(3.75\times 10^{-3}\int_0^t Z_1(s)Z_3(s)ds\right). \end{aligned} \tag{1}$$

Now use

$$\mathbb{E}f(X(t)) = \mathbb{E}[f(X(t)) - f(Z(t))] + \mathbb{E}f(Z(t)).$$

Another example: Viral infection

$$X(t) = X(0) + Y_{1,1}\left(\int_0^t \min\{X_3(s), Z_3(s)\}ds\right)\zeta_1 + Y_{1,2}\left(\int_0^t X_3(s) - \min\{X_3(s), Z_3(s)\}ds\right)\zeta_1$$
$$+ Y_{2,1}\left(0.025\int_0^t \min\{X_1(s), Z_1(s)\}ds\right)\zeta_2 + Y_{2,2}\left(0.025\int_0^t X_1(s) - \min\{X_1(s), Z_1(s)\}ds\right)\zeta_2$$
$$+ Y_3\left(1000\int_0^t X_3(s)ds\right)\zeta_3$$
$$+ Y_{4,1}\left(0.25\int_0^t \min\{X_3(s), Z_3(s)\}(s)ds\right)\zeta_4 + Y_{4,2}\left(0.25\int_0^t X_3(s) - \min\{X_3(s), Z_3(s)\}(s)ds\right)\zeta_4$$
$$+ Y_5\left(2\int_0^t X_2(s)ds\right)\zeta_5$$
$$+ Y_{6,1}\left(\int_0^t \min\{\lambda_6(X(s)), \Lambda_6(Z(s))\}ds\right)\zeta_6 - Y_{6,2}\left(\int_0^t \lambda_6(X(s)) - \min\{\lambda_6(X(s)), \Lambda_6(Z(s))\}ds\right)\zeta_6$$

$$Z(t) = Y_{1,1}\left(\int_0^t \min\{X_3(s), Z_3(s)\}ds\right)\zeta_1 + Y_{1,3}\left(\int_0^t Z_3(s) - \min\{X_3(s), Z_3(s)\}ds\right)\zeta_1$$
$$+ Y_{2,1}\left(0.025\int_0^t \min\{X_1(s), Z_1(s)\}ds\right)\zeta_2 + Y_{2,3}\left(0.025\int_0^t Z_1(s) - \min\{X_1(s), Z_1(s)\}ds\right)\zeta_2$$
$$+ Y_{4,1}\left(0.25\int_0^t \min\{X_3(s), Z_3(s)\}(s)ds\right)\zeta_4 + Y_{4,3}\left(0.25\int_0^t Z_3(s) - \min\{X_3(s), Z_3(s)\}(s)ds\right)\zeta_4$$
$$+ Y_{6,1}\left(\int_0^t \min\{\lambda_6(X(s)), \Lambda_6(Z(s))\}ds\right)\zeta_6 - Y_{6,3}\left(\int_0^t \Lambda_6(Z(s)) - \min\{\lambda_6(X(s)), \Lambda_6(Z(s))\}ds\right)\zeta_6,$$

Another example: Viral infection

Suppose want

$$\mathbb{E}X_{virus}(20)$$

Given $T(0) = 10$, all others zero.

Method: Exact algorithm with crude Monte Carlo.

Approximation	# paths	CPU Time	# updates
13.85 ± 0.07	75,000	24,800 CPU S	1.45×10^{10}

Another example: Viral infection

Suppose want

$$\mathbb{E}X_{virus}(20)$$

Given $T(0) = 10$, all others zero.

Method: Exact algorithm with crude Monte Carlo.

Approximation	# paths	CPU Time	# updates
13.85 ± 0.07	75,000	24,800 CPU S	1.45×10^{10}

Method: $\mathbb{E}f(X(t)) = \mathbb{E}[f(X(t)) - f(Z(t))] + \mathbb{E}f(Z(t))$.

Approximation	CPU Time	# updates
13.91 ± 0.07	1,118.5 CPU S	2.41×10^{8}

Exact + crude Monte Carlo used:

1. 60 times more total steps.
2. 22 times more CPU time.

Thanks!

References:

1. David F. Anderson and Desmond J. Higham, *Multi-level Monte Carlo for continuous time Markov chains, with applications in biochemical kinetics*, SIAM: Multiscale Modeling and Simulation, Vol. 10, No. 1, 146 - 179, 2012.

2. Mike B. Giles. Multi-level Monte Carlo path simulation, Operations Research, 56(3):607-617, 2008.

3. David F. Anderson and Thomas G. Kurtz, *Continuous Time Markov Chain Models for Chemical Reaction Networks*, in Design and Analysis of biomolecular circuits, Springer, 2011, Eds Heinz Koeppl et al.

Funding: NSF-DMS-1009275.