Computational Methods for Stochastic Models Arising in the Biosciences

David F. Anderson*

*anderson@math.wisc.edu

Department of Mathematics

University of Wisconsin - Madison

SIAM LIfe Sciences 2012

August 8th, 2012.

Stochastic Models of Biochemical Reaction Networks

- Stochastic models of biochemical reaction networks are often continuous time Markov chains.
- Often called chemical master equation type models.

The most common simulation methods include

1. Gillespie's Algorithm.
2. The next reaction method of Gibson and Bruck.
3. These are examples of discrete event simulation.

Stochastic Models of Biochemical Reaction Networks

Each exact method produces sample paths that can approximate values such as

$$\mathbb{E}f(X(t))$$

For example,

1. Means.
2. Variances.
3. Probabilities.
4. Hitting times.

or

$$\frac{d}{d\theta}\mathbb{E}f(X(t,\theta)).$$

Problem: solving using these exact algorithms can be computationally expensive.

Beginning of solution: build relevant mathematical representation.

Will develop Multi-level Monte Carlo.

Build up model: Random time change representation of Kurtz

Consider the simple system

$$A + B \to C$$

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

Simple book-keeping: if $X(t) = (X_A(t), X_B(t), X_C(t))^T$ gives the state at time t,

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

where

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

Build up model: Random time change representation of Kurtz

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 point process.

Hence

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

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

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

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

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

  Y_k are independent, unit-rate Poisson processes.

Today's problem

Problem: 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 estimator

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

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

What is the computational cost?

Recall,

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

Thus,

$$\text{Var}(\hat{\mu}_n) = O\left(\frac{1}{n}\right).$$

So, if we want

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

we need

$$\frac{1}{\sqrt{n}} = O(\epsilon) \implies n = O(\epsilon^{-2}).$$

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

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

Can be bad if (i) $\overline{N}$ is large, or (ii) ϵ is small.

Benefits/drawbacks

Benefits:

1. Easy to implement.
2. Estimator

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

   is unbiased and consistent.

Drawbacks:

1. The cost of $O(\overline{N}\epsilon^{-2})$ 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 [1] or Euler's method, was first formulated by Dan Gillespie in this setting .

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

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

[1] 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

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

We note

$$\mathbb{E}f(X(t)) - \hat{\mu}_n = \left[\mathbb{E}f(X(t)) - \mathbb{E}f(Z_L(t))\right] + \left[\mathbb{E}f(Z_L(t)) - \hat{\mu}_n\right]$$

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 $\epsilon^{-1} \ll \overline{N}$.

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

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

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/consistent estimator.

Multi-level Monte Carlo and Control variates

- Suppose I want

$$\mathbb{E}X \approx \frac{1}{n}\sum_{i=1}^{n} X_{[i]},$$

but realizations of X are expensive.

- Suppose $X \approx \widehat{Z}_L$, and $\widehat{Z}_L$ is cheap.

- Suppose $X, \widehat{Z}_L$ can be generated simultaneously so that

$$\mathrm{Var}(X - \widehat{Z}_L)$$

is small.

- Then use

$$\mathbb{E}X = \mathbb{E}[X - \widehat{Z}_L] + \mathbb{E}\widehat{Z}_L \approx \frac{1}{n_1}\sum_{i=1}^{n_1}(X_{[i]} - \widehat{Z}_{L,[i]}) + \frac{1}{n_2}\sum_{i=1}^{n_2}\widehat{Z}_{L,[i]}.$$

- Keep going

$$\mathbb{E}X = \mathbb{E}(X - \widehat{Z}_L) + \mathbb{E}\widehat{Z}_L = \mathbb{E}(Z - \widehat{Z}_L) + \mathbb{E}(\widehat{Z}_L - \widehat{Z}_{L-1}) + \mathbb{E}\widehat{Z}_{L-1} = \cdots$$

Multi-level Monte Carlo: an unbiased estimator

In our setting:

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

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

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

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

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

and note that

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

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

So what is coupling and the variance?

The coupling

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

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

Now couple

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

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

Multi-level Monte Carlo: chemical kinetic setting

Can prove:

Theorem (Anderson, Higham 2012)

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 (Anderson, Higham 2012)

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

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

---

[1] David F. Anderson and Desmond J. Higham, *Multi-level Monte Carlo for stochastically modeled chemical kinetic systems*, 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

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

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 and 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!)	8.27×10^{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}

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	1.1×10^{9}
$M = 3$, $L = 5$	3,714.7 $\pm$ 1.0	1,114.9 S	9.4×10^{8}
$M = 4$, $L = 5$	3,713.8 $\pm$ 1.0	1,014.9 S	1.1×10^{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).
3. Will be at worst the same speed as Gillespie (exact algorithm + crude Monte Carlo).
4. Will commonly be many orders of magnitude faster.
5. Applicable to essentially *all* continuous time Markov chains:

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

6. 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:

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

- 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.

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$(500 \times T)$.

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

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

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

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 stochastically modeled chemical kinetic systems*, SIAM: Multiscale Modeling and Simulation, Vol. **10**, No. 1, 146 - 179, 2012.

2. David F. Anderson and Thomas G. Kurtz, *Continuous time Markov chain models for chemical reaction networks*, chapter in Design and Analysis of Biomolecular Circuits: Engineering Approaches to Systems and Synthetic Biology, H. Koeppl et al. (eds.), Springer, 2011.

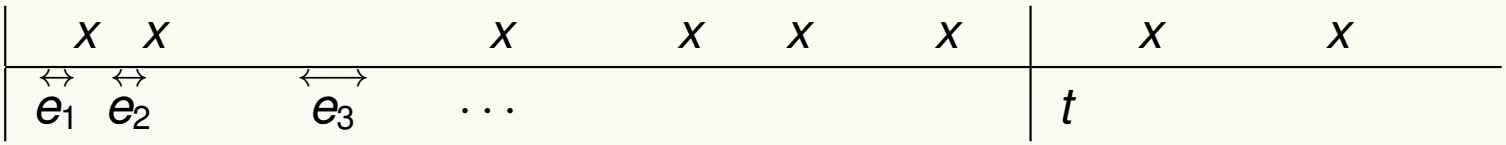
Funding: NSF-DMS-1009275.

Background information: The Poisson process

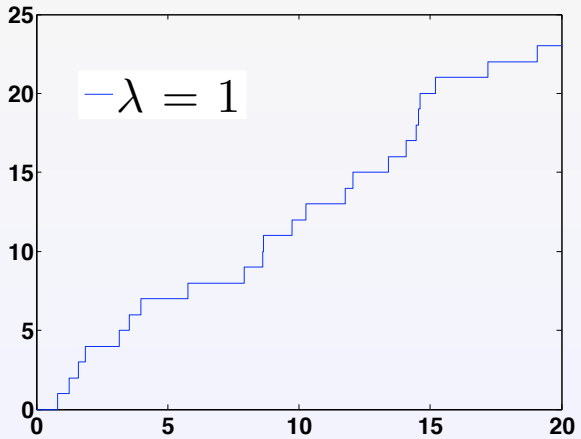
A Poisson process, $Y(\cdot)$, is a model for a series of random observations occurring in time.

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

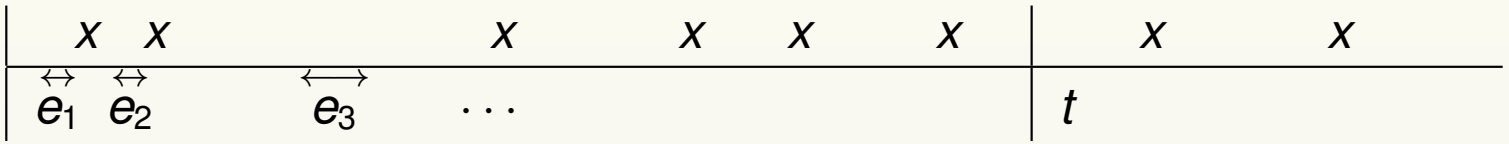


The Poisson process

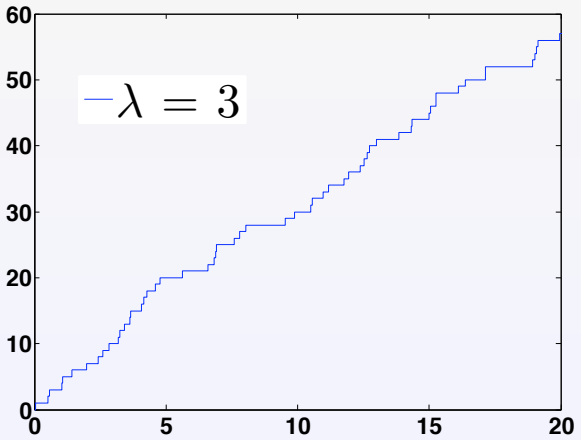
Let

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

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 an nonhomogeneous 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.

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.

Biological 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.

Models of interest

$$A + B \to C$$

Simple book-keeping: if $X(t) = (X_A(t), X_B(t), X_C(t))$ gives the state at time t,

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

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

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

and that for an inhomogeneous Poisson process with rate $\lambda(t)$ we have

$$P\{Y_\lambda(t+\Delta t) - Y_\lambda(t) > 0 | \mathcal{F}_t\} = 1 - \exp\left\{-\int_0^{\Delta t} \lambda(t) dt\right\} \approx \lambda(t)\Delta t.$$

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 $t \geq 0$.