Random walks, Birth Death Processes, and the Gillespie Algorithm. [1]

Pankaj Mehta

March 29, 2019

In this lecture, we begin our foray into stochastic processes. We discuss random walks and simple birth death processes before moving on to one of the simplest simulation algorithms for simulating chemical kinetics: the Gillespie Algorithm. This serves as gentle introduction to numerical simulations of stochastic processes.

[1] For simple introduction to Gillespie read CV Rao, AP Arkin - The Journal of chemical physics 118:4999 (2003). The best introductory book on stochastic processes is Van Kampen *Stochastic Processes in Physics and Chemistry*. Much of these notes are based on unpublished chapter in Anirvan Sengupta's *Modeling Molecular Networks*. Also worth reading Gillespie's original paper from 1977.

Random Walks and Diffusion

Consider a particle moving in one dimension on a lattice ("Be wise, discretize!" as M. Kac is supposed to have advised). The lattice sites are are specified by integers. Assume that at every discrete time step a particle moves either to the left or to the right with equal probability. If the particle starts at position 0, what is the probability distribution of its positions after N such time steps? The mean displacement is of course zero since the particle is equally likely to go left or right. However, from experience we know that the particle is actually likely to end us quite far from the origin on any given realization. In this section, we introduce the mathematics to describe such processes.

A random walk can be described as an N letter word of the form "LRLRRRL...R" where a L or R at the i-th position indicates that the i-th move is to the left or to the right, respectively. If after N steps, we have l left moves and $r = N - l$ right moves, our total displacement is $q = r - l = N - 2l$. The number of moves with a fixed l (and, therefore, a fixed displacement q) is the combinatorial factor ${}^{N}C_l = N!/(l!(N-l)!)$. Since there are a total of 2^N possible walks, the probability of having a displacement q after N steps is just

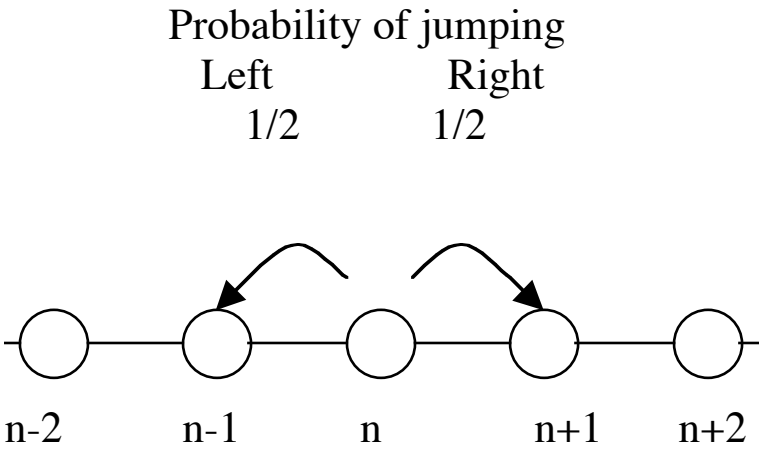
$$P_{q,N} = \frac{{}^{N}C_l}{2^N} = \frac{N!}{l!r!}\frac{1}{2^N} = \frac{N!}{(\frac{N+q}{2})!(\frac{N-q}{2})!}\frac{1}{2^N} \tag{1}$$

Note that q only takes odd or even values depending upon whether number of steps N is odd or even.

To simplify this equation, we make use of the Stirling approximation to the factorial for large M,

$$\ln M! = M\ln M - M + \frac{1}{2}\ln(2\pi M) + O(\frac{1}{M}) \tag{2}$$

Figure 1: Random walk on a lattice



and $1 \gg m \gg M$,

$$\begin{aligned}\ln(M+m)! &= (M+m)\ln(M+m) - (M+m) + \frac{1}{2}\ln(2\pi(M+m)) + O(\frac{1}{M}) \\ &= M\ln M - M + \frac{1}{2}\ln(2\pi M) + m\ln M + \frac{1}{2}\frac{m^2}{M} + O(\frac{m}{M}). \qquad (3)\end{aligned}$$

Using the equations 2 and 3 to expand the factorial in Eq. 1 and noting $1 \ll q \ll N$, we have

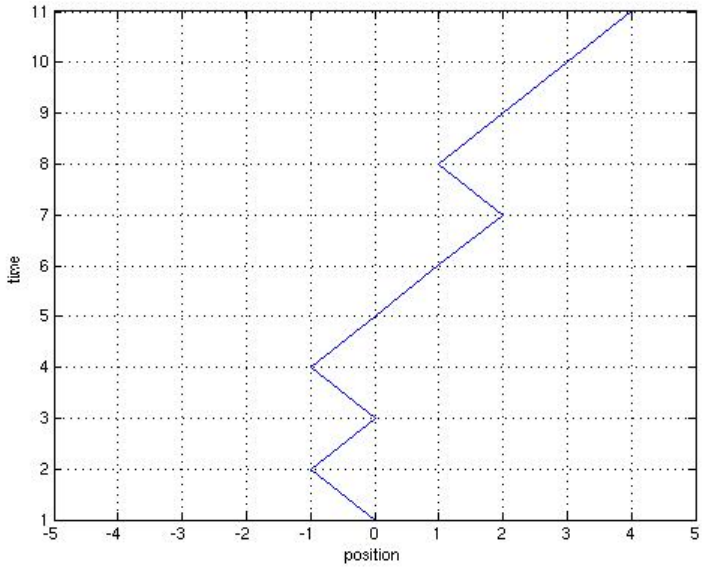
$$P_{q,N} = \frac{N!}{(\frac{N+q}{2})!(\frac{N+q}{2})!}\frac{1}{2^N} \approx \frac{1}{\sqrt{2\pi N}}e^{-\frac{q^2}{2N}} \times 2 \qquad (4)$$

This is just the gaussian approximation to a binomial distribution. Thus,

$$\begin{aligned}\text{Prob}[a \le q \le b] &= \sum_{q\in\{a,a+2,\ldots,b-2,b\}} P_{q,N} \\ &\approx \sum_{q\in\{a,a+2,\ldots,b-2,b\}} \frac{1}{\sqrt{2\pi N}}e^{-\frac{q^2}{2N}} \times 2 \\ &\approx \int_a^b \frac{dq}{\sqrt{2\pi N}}e^{-\frac{q^2}{2N}}, \qquad (5)\end{aligned}$$

where we have used the fact that as the lattice size goes to zero sums can be replaced by integrals and changing l by 1 changes q by a factor of 2. Coming back to continuous space and time, let us have the

Figure 2: An example of a random walk



lattice spacing to be Δx and time steps to be Δt. Then $x = q\Delta x$ and $t = N\Delta t$. The approximate probability distribution of x is then written as

$$p(x,t) \approx \frac{1}{\sqrt{2\pi N}} e^{-\frac{q^2}{2N}} \frac{dq}{dx} = \frac{1}{\sqrt{4\pi Dt}} e^{-\frac{x^2}{4Dt}}, \tag{6}$$

where we have defined the diffusion constant $D = \Delta x^2/(2\Delta t)$. To see that this definition is consistent with the usual definition of D, we can directly derive the evolution equation for $p(x,t)$ starting with the recursion relation

$$P_{q,N+1} = \frac{1}{2}[P_{q-1,N} + P_{q+1,N}]. \tag{7}$$

Since $P_{q,N}$ is proportional to $p(x,t)$ for $x = q\Delta x$ and $t = N\Delta t$

$$p(x,t+\Delta t) = \frac{1}{2}[p(x-\Delta x,t) + p(x+\Delta x,t)] \tag{8}$$

or

$$\begin{aligned} p(x,t) + \Delta t \partial_t p(x,t) + O(\Delta t^2) &= \frac{1}{2}[p(x-\Delta x, t+\Delta t) + p(x+\Delta x, t+\Delta t)] \\ &= \frac{1}{2}[p(x,t) - \Delta x \partial_x p(x,t) + \frac{1}{2}\Delta x^2 \partial_x^2 p(x,t) \\ &\quad + p(x,t) + \Delta x \partial_x p(x,t) + \frac{1}{2}\Delta x^2 \partial_x^2 p(x,t)] + O(\Delta x^4) \\ &= p(x,t) + \frac{1}{2}\Delta x^2 \partial_x^2 p(x,t) + O(\Delta x^4) \end{aligned} \tag{9}$$

implying

$$\partial_t p(x,t) \approx \frac{\Delta x^2}{2\Delta t} \partial_x^2 p(x,t) = D\partial_x^2 p(x,t). \tag{10}$$

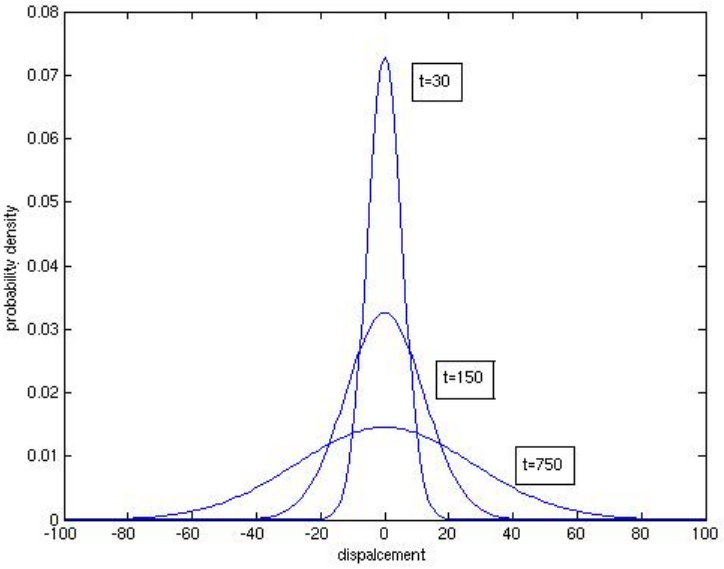


Figure 3: Gaussian distribution for different values of time.

This is known as the Fokker-Planck equation.

If we have many particles with positions x_i, the average density $\rho(x,t) = \sum_i \langle \delta(x - x_i(t)) \rangle = p(x,t) \times$ Number of particles. Thus $\rho(x,t)$ also satisfies equation 10. Thus we derive the diffusion equation in one dimension, with D identified as the diffusion constant. We leave the generalization to higher dimensions as an exercise.

Excercise: Check that $p(x,t) = \exp(-x^2/(4\pi Dt))/\sqrt{4\pi Dt}$ is the solution of equation 10 with the initial condition $p(x,0) = \delta(x)$.

Excercise: Consider that case of a bias one-dimensional random walker where a particle can hop to the left with probability $1/2 + b$ and to the right with $1/2 - b$, with $0 < b < 1/2$.

a) Write down the evolution equation for $P_{q,N+1}$ and $p(x,t)$.

b) Derive the appropriate diffusion equation.

Birth-Death Processes

Before moving on to thinking about stochastic aspects of chemical kinetics and gene regulation, it worth considering a simple class of processes known as birth-death processes. Birth-death processes describe many systems of interest in biological physics including chemical kinetics, ecology, and even evolutionary processes. They

also serve as a wonderful playground for learning about probability, stochasticity, and non-equilibrium statistical mechanics.

Birth-death with a single species/molecule type

Consider a system with a single species/molecule type. Let us denote the number of molecules in the system by n. A molecules can be created (birthed) at a rate $f(n)$ per unit time and destroyed (die) at a rate $g(n)$ per unit time. Notice that in general these rates depend on the number of molecules in the system. For example, if each molecule/individual dies at a rate τ^{-1}, then $g(n) = \tau^{-1}n$.

Such a process can be described by a *Master Equation*. Let us denote the probability of having n molecules at time t by $p(n,t)$. Then the dynamics of such a process is given by the equation

$$\frac{dp(n,t)}{dt} = f(n-1)p(n-1,t) + g(n+1)p(n+1,t) - f(n)p(n,t) - g(n)p(n,t). \tag{11}$$

The first term on the right-hand side describes the probability that the system has $n-1$ particles and a new particle is birthed in a time dt. The second term describes the probability that the system has $n+1$ particles and a particle is destroyed. The third and fourth terms describe the probability that the system has n particles and a new particle is created and destroyed respectively.

In general, this kind of Master Equation is extremely difficult to solve. To get more intuition, we can ask about the behavior of the mean-number of particles $\langle n \rangle$. This is the "deterministic" behavior where we ignore all fluctuations. To derive this, we can use the Master equation to write down an ODE that describes the behavior of this mean

$$\begin{aligned}
\frac{d\langle n\rangle}{dt} &= \frac{d\sum_n p(n,t)n}{dt} = \sum_n \frac{dp(n,t)}{dt}n \\
&= \sum_n n\left(f(n-1)p(n-1,t) + g(n+1)p(n+1,t) - f(n)p(n,t) - g(n)p(n,t)\right) \\
&= \sum_n{}' (n'+1)f(n')p(n',t) + \sum_m (m-1)g(m)p(m,t) - \sum_n n(f(n)+g(n))p(n,t) \\
&= \langle f(n)\rangle - \langle g(n)\rangle,
\end{aligned} \tag{12}$$

where in going from the second to third line we have written $n' = n-1$ and $m = n+1$. This is of course the usual kinetic equations we studied earlier in the class.

General Birth-Death Processes

So far we have only described a process in which a molecule is only made not destroyed. In biological systems many molecules have ded-

icated enzymes for destroying them. RNA and proteins are degraded by RNases and proteases, respectively, and both play important roles in gene expression regulation. Proteins like phosphodiesterase convert cyclic nucleotide monophosphate to nucleotide monophosphate, and affect signaling. For any posttranslational modification of proteins, like phosporylation etc., there are enzymes like posphatases, that undo the change.

In general, for each molecule the birth rate and the death rate can depend on the number of other molecules present. We specify the state of the cell by a vector of numbers for the different species of molecules, $\vec{n} = (n_1, n_2, ..., n_k)$, the rate of synthesis of species i, $(n_1, n_2, ..., n_i, ..., n_k) \rightarrow (n_1, n_2, ..., n_i + 1, ..., n_k)$, by $f_i(\vec{n})$, and the rate of degradation of species i,$(n_1, n_2, ..., n_i, ...n_k) \rightarrow (n_1, n_2, ..., n_i - 1, ..., n_k)$, by $g_i(\vec{n})$.

Gillespie Algorithm

Before proceeding to analytic approximations, it is useful to discuss how to numerically simulate the chemical reactions like birth-death processes. One approach one can imagine to simulating these reactions is to choose a small time step $\Delta t \ll 1$, draw a uniform random number for each reaction, check if a synthesis or degradation event occurs during the time step by determining if the corresponding random number is smaller than $f_i(\vec{n})\Delta t \ll 1$ or $g_i(\vec{n})\Delta t \ll 1$, updating the state of the cell, and then repeating the process. The problem with such a naive approach is that since the probability of an event occurring in any time step is extremely small. In fact, during most time steps nothing will happen. Consequently, such simulations are extremely inefficient and slow. One can imagine speeding up the simulation by increasing Δt. However, for larger Δt one quickly runs into the problem that there is a non-zero probability of having multiple events during each time step. . An alternative approach, often termed the "Gillespie Algorithm", circumnavigates the problems discussed above and has quickly become the standard technique for simulating stochastic chemical reactions in systems biology. We now discuss how to use the Gillespie algorithm to simulate an arbitrary set of chemical reactions. As before, denote the number of molecules present of all species by $\vec{n}$. Furthermore, index the possible reactions by P, with the rate of reaction P, $\vec{n} \rightarrow \vec{n} + \vec{e}_P$, given by $r_P(\vec{n})$. For example for the the birth-death processes discussed above, we can consider the reaction for the creation of a molecule of species i. For this case, $r_P = f_i(\vec{n})$ we have that $e_p = (0, 0, \ldots, 1, \ldots, 0)$, the vector with 1 at the i-th position and zero everywhere else. The key observation behind the Gillespie algorithm is that each reaction is an

independent Poisson process so we can explicitly calculate the waiting time distribution between events. In particular, the probability that *any* event occurs is a Poisson process with rate $R = \sum_P r_P$. The reason for this is that the sum of independent Poisson processes is itself a Poisson process

Exercise: We will make use of a number of basic properties of Poisson processes that we will prove in this exercise

- Prove that the sum of N independent Poisson processes with rates $r_1, r_2, \ldots, r_N$ is another Poisson process with rate $r_{tot} = \sum_{j=1}^{N} r_j$.
- We can define a random variable $\tau \in (0, \infty)$ which measures the time until the next event occurs. We will now ask about the probability that event j occurs in this time interval. Show that the probability that an even occurs exactly in the interval $(\tau, \tau + dt)$ is given by $P(\tau, j) = e^{-r_{tot}\tau} r_j dt$.
- Show that we can write this probability at $P(\tau, j) = P(j|\tau)P(\tau)$ with
$$P(\tau) = e^{-r_{tot}\tau} r_{tot} dt \tag{13}$$
the time until the next event and
$$P(j|\tau) = \frac{r_j}{r_{tot}}, \tag{14}$$
the probability that reaction j occurs.

We now have to think about how we sample τ. From the exercise above, we know that τ is an exponentially distributed variable. Notice that the τ are drawn from an exponential distribution [2] How can we sample τ? The key idea is to use what is called *inverse transform sampling*. Imagine we want to sample some random variable Y. Furthermore, imagine that we know that there exists an easily invertible function C such that $C(Y) = U$, where U is a uniform random variable U on the unit interval $[0, 1]$. Then, we can sample Y by drawing a random number U (which is easy to do), and calculating $Y = C^{-1}(U)$.

[2] In general, whenever a new distribution is introduced, I encourage people to read about it on Wikipedia.

Naively, one might think that it is extremely difficult to find such a function C. However, its actually not as hard as one might think. Imagine, that the random variable is drawn from some distribution $p(Y)$. We can than define a cumulative distribution

$$C(Y) = \int_{-\infty}^{Y} dY' p(Y') \tag{15}$$

Notice that if we define a random variable $U = C(Y)$, by definition it is uniform on the unit variable. After all, this is the definition of a cumulative distribution. If we can easily invert the function C analytically, we are done.
Exercise: Use the exercise above to define a sampling scheme for the waiting time τ for a Poisson process.

These observations and exercise above suggests the following algorithm for simulating chemical reactions:

- Initialize the simulation at some $\vec{n} = \vec{n}_0$ and time $t = 0$.
- Draw a random number x_1 uniformly distributed between 0 and 1.
- Explicitly calculate the waiting time τ between event using $\tau = -\log(x_1)/r$. This step uses the usual Monte-Carlo sampling procedure based on the cumulative distribution function (cdf) of a Poisson process, $P(t)$.
- Draw a second random number x_2 uniformly distributed between 0 and 1 to choose which of the reactions occurs. Reaction P occurs if
$$\frac{\sum_{j=1}^{p-1} r_j}{R} \leq x_2 < \frac{\sum_{j=1}^{p} r_j}{R} \tag{16}$$
.
- Update the time, $t \to t + \tau$, and $\vec{n}$ using appropriate reaction.

We end this section by emphasizing that the Gillespie algorithm outlined above is exact. No approximations of any kind were utilized. For this reason, the Gillespie algorithm has become one of the workhorses of simulating biochemical reactions

Homework

Excercise: Use the Gillespie algorithm to simulate a simple birth-death process for a single species where molecules are synthesized at a rate $f(n) = R$ and are degraded at a rate $g(n) = \tau^{-1}n$.

Excercise: Use the Gillespie algorithm to simulate a stochastic version of the following coupled birth death processes.

-
$$\frac{dn_1}{dt} = \alpha - \beta n_1 \tag{17}$$
$$\frac{dn_2}{dt} = \alpha_2 \frac{n_1^j}{K + n_1^j} - \beta_2 n_2 \tag{18}$$

for $\alpha = 1$, $\beta = 0.5$, $\alpha_2 = 5$, $\beta_2 = 1/30$, $K = 1$. Interpret your results. Change $K = 5$. How do your results differ.

- Calculate the mean and variance of the distributions. Calculate the Fano factor $\sigma_n^2/\bar{n}$ for the two species. Can you give a simple explanation for your results?