Advanced Topics in Theoretical Neuroscience, Spring 2009: Synchrony and Stochastic Effects in Neural Networks and beyond

Jack Cowan, with Edward Wallace and Marc Benayoun

Lecture 6, April 16th, 2009

Plan: Exact Stochastic Simulation

- The exponential distribution quickly reviewed
- Continuous-time discrete-state Markov processes reviewed
- The Gillespie Algorithm Idea
- Gillespie in action
- continuous approximation via τ-leaping,

The exponential distribution in a nutshell

Suppose you have an event taking place (e.g. buses arriving or infections starting) at a constant rate μ. This means that,

$$\mathcal{P}(\text{an event happens in interval } (t, t+dt)) = \mu dt + o(dt^2) \quad (1)$$

so, suppose $t = n\,dt$, and then

$$\mathcal{P}(\text{first event is in interval } (t, t+dt)) = (1-\mu dt+o(dt^2))^n(\mu dt+o(dt^2)) \quad (2)$$

and taking the limit $n \to \infty$, $n\,dt = t$, we obtain the exponential distribution

$$\mathcal{P}(\text{first event happens before time } t) = \int_0^t \mu e^{-\mu t'} dt' = 1 - e^{-\mu t} \quad (3)$$

An exponential distribution has distribution function $\mu e^{-\mu t}$. By similar reasoning, if you have n possible events, each happening randomly at constant rate μ_i, then the distribution function of the first event is $\sum_i \mu_i e^{-\sum_i \mu_i t}$.

Randomly generating samples from an exponential distribution

The basic idea is that, for a continuously distributed random variable $T > 0$ the cumulative distribution function

$$F(t) = \int_0^t f(t')dt' \tag{4}$$

must by definition have

$$F(T) \sim U(0,1) \tag{5}$$

uniformly distributed. So if F is invertible, i.e. if $f(t) > 0$ for $t > 0$, and $X \sim U(0,1)$, then

$$F^{-1}(X) \tag{6}$$

has the same distribution as T.

So, if we want to generate an exponential random variable, we take

$$T \sim F^{-1}(X) \sim -\frac{\log(1-X)}{\mu} \sim -\frac{\log(X)}{\mu} \tag{7}$$

since $1 - X \sim X \sim U(0,1)$.

Continuous-time autonomous Markov processes

Now consider a random process with states $1, 2, 3, \ldots$, where

$$P(x_{n+1}, t_{n+1}|x_n, t_n; \ldots x_0, t_0) = P(x_{n+1}, t_{n+1}|x_n, t_n) \qquad \text{Markov} \tag{8}$$

$$P(x_{n+1}, t_{n+1}|x_n, t_n) = P(x_{n+1}, t_{n+1} - t_n|x_n, 0) \qquad \text{autonomous} \tag{9}$$

Then the transition rates are constant in time,

$$\lambda_{ij} = \lim_{dt \to 0} \frac{1}{dt} P(i, dt|j, 0) \qquad \lambda_j = \sum_i \lambda_{ij} \tag{10}$$

giving the master equation

$$\dot{p}_j(t) = \sum_k \lambda_{jk} p_k(t) - \sum_i \lambda_{ij} p_j(t) = \sum_k \lambda_{jk} p_k(t) - \lambda_j p_j(t) \tag{11}$$

By the reasoning on the previous slide, the time spent in the state j prior to a transition is exponentially distributed with parameter λ_j. Moreover, because the transitions are independent of each other, the probability

$$\mathcal{P}(j \to i| \text{ jumps at time } t) = \frac{\lambda_{ij}}{\lambda_j} \tag{12}$$

as can be easily seen by considering dt small.

The Gillespie Algorithm Idea

The above considerations suggest directly an algorithm to simulate continuous-time discrete state Markov processes, starting in state j:

1. Find the individual transition rates λ_{ij}, and transition rate out of state j $\lambda_j := \sum_i \lambda_{ij}$.
2. Pick time increment dt from an exponential distribution of rate λ_j, i.e $dt = -\frac{\log(U(0,1))}{\lambda_j}$.
3. Pick ith state with probability $\frac{\lambda_{ij}}{\lambda_j}$, change to ith state, and update time to $t + dt$.

Dan Gillespie wrote this down in 1976 and called it the stochastic simulation algorithm. It is *exact*. Inexactness can only come from rounding errors in calculating

- $\frac{\log(U(0,1))}{\lambda_j}$, if some λ_j are very small; or
- $\frac{\lambda_{ij}}{\lambda_j}$, if some λ_{ij} are much, much bigger than others for fixed j.

Gillespie in Action

τ-leaping

On board from Gillespie 2003