

Chapter 2: Modeling Enzyme Kinetics

From a mathematical point of view, the art of good modeling relies on: (i) a sound understanding and appreciation of the biological problem; (ii) a realistic mathematical representation of the important biological phenomena; (iii) finding useful solutions, preferably quantitative; and most crucially important, (iv) a biological interpretation of the mathematical results in terms of insights and predictions. The mathematics is dictated by the biology and not vice versa. Sometimes the mathematics can be very simple. Useful mathematical biology research is not judged by mathematical standards but by different and no less demanding ones.

- Jim Murray, 1993

Introduction

When investigating a novel method, it is often very useful to use a small example, or “toy problem,” to examine its workings before jumping into a larger problem. As mentioned previously, these simplified problems are hard to come by in biology but there is a toy problem at the heart of the simulation, namely, enzymatic reactions. This chapter contains a description of the basic enzyme reaction first described by Michaelis and Menten in 1913, as well as a comparison between the results of the deterministic solution and the stochastic solution. This problem was chosen for a variety of reasons. Firstly, this problem contains much of the basics of enzymatic biology in its midst. Secondly,

this is one of the few problems for which a detailed solution can be constructed. Finally, it affords a readily understandable introduction to the stochastic simulation.

In the case of the deterministic solution, perturbation theory will be used to provide an approximate solution, but because any modern computer algebra system will be able to easily provide a numerical solution to the resulting differential equations, that will be included as well. To solve this problem using a stochastic solution, a short program using the *Mathematica* programming language has been developed. This will allow a detailed example of the stochastic simulation algorithm. By comparing the solutions from the deterministic methods to the stochastic simulation solutions, it will be shown that they are in good agreement with each other in a global sense. However, it will also be shown that there are situations in which the deterministic solution may not capture the true state of the system.

Deterministic Solution

The theory for chemical kinetics in a large volume is well grounded in experiments. Early forms of the Law of Mass Action, which states that the rate of the reaction is proportional to the concentration of the reactants, appeared at least as early as 1802 with Berthollet's nearly correct formulations. The final correct formulation came from extensive experiments that Waage and Gulberg published in 1864 (Waage, 1986). But an important piece of the puzzle was still missing, and it was another 25 years before the discovery of the process that allowed some molecules to react while others remained inactive.

In 1889, while investigating an offshoot of his work on ionic solutions (work that would eventually win him a Nobel Prize), Svante Arrhenius studied the effects temperature has on the rate of a reaction. His data led him to conclude that in a reaction system only a certain number of molecules are able to react at any given time. He proposed that some sort of chemical catalyst must have activated the molecules that are able to react. His theory said that the catalyst (C) would first form an intermediate compound (CS) with the substrate (S), and the resulting compounds are then able to enter a transition state that lowers the amount of energy that is needed to perform a chemical reaction. The compound then decomposes into a product (P) and the catalyst (C), and the catalyst is then free to participate in another reaction:



Thus, the notion of activation energy for a chemical reaction was born (Teich, 1992).

Two decades later, Michaelis and Menten published a seminal piece of work on how this type of system behaves. In their paper, they focused on a biological system that has come to be known as the basic enzyme reaction (Michaelis and Menten, 1913). It was very similar to Arrhenius system but with the addition of a backwards reaction (disassociation) of the complex (ES). There was also a terminology change from catalyst (C) to enzyme (E). This was just a minor change as an enzyme is defined to be an organic catalyst. Schematically, this can be represented by



In words, one molecule of the enzyme combines with one molecule of the substrate to form one molecule of the complex. The complex can disassociate into one molecule of

each of the enzyme and substrate, or it can produce a product and a recycled enzyme. In this formulation k_1 is the rate parameter for the forward substrate/enzyme (catalyst), k_{-1} is the rate parameter for the backwards reactions, and k_2 is the rate parameter for the creation of the product. There is no backwards reaction forming the complex from the product and the enzyme, as it is assumed that this reaction is energetically unfavorable and the enzyme is much more likely to participate with the substrate in the formation of the complex. Given an initial amount (or concentrations) of the reactants and the rate parameters, the question is to determine the amount of product at some later time.

Using the Law of Mass Action, it is possible to write down the change in the amount of each of the reactants, leading to one differential equation for each of the reactants. The fact that it may not adequately capture the true state of small systems is a problem that will be addressed shortly. The presentation of the basic enzyme reaction that follows draws from the conventional approaches (Edelstein-Keshet, 1998; Murray, 1993).

Denoting the concentrations in (2.2) by

$$e = [E], s = [S], c = [ES], p = [P] \quad (2.3)$$

the Law of Mass Action applied to this system leads to the following four differential equations that describe the kinetics of the basic enzyme reaction:

$$\begin{aligned} \frac{ds}{dt} &= -k_1 es + k_{-1} c, & \frac{de}{dt} &= -k_1 es + (k_{-1} + k_2) c \\ \frac{dc}{dt} &= k_1 es - (k_{-1} + k_2) c, & \frac{dp}{dt} &= k_2 c \end{aligned} \quad (2.4)$$

As the system starts with only the substrate and enzymes, the initial conditions are then

$$e(0) = e_0, s(0) = s_0, c(0) = 0, p(0) = 0. \quad (2.5)$$

Before solving this system, it is important to note that the equations are not all independent. First of all, given a fixed amount of enzyme, it is possible to write down a conservation law by noting that the amount of free enzyme and bound enzyme must be constant:

$$e(t) + c(t) = e_0. \quad (2.6)$$

Combining this back into the first three differential equations, it is possible to eliminate one to end up with

$$\frac{ds}{dt} = -k_1 e_0 s + (k_1 s + k_{-1} c), \quad \frac{dc}{dt} = k_1 e_0 s - (k_1 s + k_{-1} + k_2) c \quad (2.7)$$

with the initial conditions

$$s(0) = s_0, c(0) = 0. \quad (2.8)$$

Finally, the equation for the product can be uncoupled from the others, and integration leads to

$$p(t) = k_2 \int_0^t c(u) du, \quad (2.9)$$

which provides the solution for the product once the solution for the complex is known.

The end result is a reduction of the set of four differential equations into two coupled ones.

As the situation under consideration is one where there are a small number of enzymes compared to the number of substrate molecules available, let

$$\varepsilon = \frac{e_0}{s_0}, \quad (2.10)$$

which leads to using the following variables to nondimensionalize the equations

$$\tau = k_1 e_0 t, \quad u(\tau) = \frac{s(t)}{s_0}, \quad v(\tau) = \frac{c(t)}{e_0}, \quad \lambda = \frac{k_2}{k_1 s_0}, \quad K = \frac{k_{-1} + k_2}{k_1 s_0}. \quad (2.11)$$

Then the system in (2.7) becomes

$$\frac{du}{d\tau} = -u + (u + K - \lambda)v, \quad \varepsilon \frac{dv}{d\tau} = u - (u + K)v, \quad (2.12)$$

with the initial conditions

$$u(0) = 1, \quad v(0) = 0. \quad (2.13)$$

In looking for a solution for this problem, the appearance of the small parameter ε in front of a derivative in (2.12) suggests that this is a singular perturbation problem, and looking for a single regular Taylor series expansion solution in terms of the variables u, v and ε will not be fruitful. Because of this, it is necessary to create a multiscale solution from matching inner and outer solutions. This can be accomplished by first looking for the regular Taylor expansion solution in the form

$$u(\tau; \varepsilon) = \sum_{n=0} \varepsilon^n u_n(\tau), \quad v(\tau; \varepsilon) = \sum_{n=0} \varepsilon^n v_n(\tau). \quad (2.14)$$

Substituting this into (2.12) and equating like powers of ε yields for the $O(1)$ system

$$\frac{du_0}{d\tau} = -u_0 + (u_0 + K - \lambda)v_0, \quad 0 = u_0 - (u_0 + K)v_0 \quad (2.15)$$

with the initial conditions

$$u_0(0) = 1, \quad v_0(0) = 0. \quad (2.16)$$

At this point the problem with this type of solution is clear; the second equation does not satisfy the initial condition. This will be taken care of later when the outer and inner solutions to the system are matched. Plunging ahead and solving this system leads to

$$v = \frac{u_0}{u_0 + K}, \frac{du_0}{d\tau} = -\lambda \frac{u_0}{u_0 + K}, \quad (2.17)$$

and therefore

$$u_0(\tau) + K \ln(u_0(\tau)) = A - \lambda\tau, v_0(\tau) = \frac{u_0(\tau)}{u_0(\tau) + K}. \quad (2.18)$$

In searching for an inner solution, define

$$\sigma = \frac{\tau}{\varepsilon}, U(\sigma; \varepsilon) = u(\tau; \varepsilon), V(\sigma; \varepsilon) = v(\tau; \varepsilon) \quad (2.19)$$

then with these transformations the system in (2.12) becomes

$$\frac{dU}{d\sigma} = -\varepsilon U + \varepsilon(U + K - \lambda)V, \frac{dV}{d\sigma} = U - (U + K)V \quad (2.20)$$

with the initial conditions

$$U(0) = 1, \quad V(0) = 0. \quad (2.21)$$

The system no longer has the small parameter ε multiplying a derivative term, and therefore it is possible to look for a solution in terms of a regular perturbation expansion

$$U(\sigma; \varepsilon) = \sum_{n=0} \varepsilon^n U_n(\sigma), V(\sigma; \varepsilon) = \sum_{n=0} \varepsilon^n V_n(\sigma). \quad (2.22)$$

Substituting this expansion into (2.20) and setting $\varepsilon = 0$ yields the $O(1)$ system

$$\frac{dU_0}{d\sigma} = 0, \frac{dV_0}{d\sigma} = U_0 - (U_0 + K)V_0, \quad (2.23)$$

with the initial conditions

$$U_0(0) = 1, \quad V_0(0) = 0. \quad (2.24)$$

The solutions of this inner system are then found to be

$$U_0(\sigma) = B, V_0(\tau) = \frac{B}{B+K} + C \exp[-\tau(K+B)] \quad (2.25)$$

At this point, all that is left is to match the solutions. Using the initial conditions and requiring that

$$\lim_{\sigma \rightarrow \infty} U_0(\sigma) = \lim_{\tau \rightarrow 0} u_0(\tau) \quad \text{and} \quad \lim_{\sigma \rightarrow \infty} V_0(\sigma) = \lim_{\tau \rightarrow 0} v_0(\tau) \quad (2.26)$$

results in $A = 1, B = 1, C = \frac{-1}{1+K}$. The resulting multiscale solution then correctly matches

as the respective limits are

$$\lim_{\sigma \rightarrow \infty} U_0(\sigma) = \lim_{\tau \rightarrow 0} u_0(\tau) = 1 \quad \text{and} \quad \lim_{\sigma \rightarrow \infty} V_0(\sigma) = \lim_{\tau \rightarrow 0} v_0(\tau) = \frac{1}{1+K}. \quad (2.27)$$

The $O(1)$ solution to the inner system is

$$U_0(\sigma) = 1, V_0(\tau) = \frac{1 - \exp[-(1+K)\sigma]}{1+K}, \quad (2.28)$$

while the $O(1)$ solution to the outer system is

$$u_0(\tau) + K \ln(u_0(\tau)) = 1 - \lambda\tau, v_0(\tau) = \frac{u_0(\tau)}{u_0(\tau) + K}. \quad (2.29)$$

In practice it is extremely difficult, if not impossible, to construct even approximate solutions to a system that contains any more reactions than the Michaelis-Menten problem and numerical methods must be used (McQuarrie, 1967).

As shown above, the Law of Mass Action applied to the basic enzyme reaction leads to a set of coupled differential equations that can be approximated using perturbation theory, and the differential equations are easily solved numerically as well. Because the Law of Mass Action is not only well grounded in experiments but also leads to equations that can be readily solved. But while differential equations are a natural way

to model chemical reactions in a vat, they might not adequately represent the true state of the system in a cell.

Implicit in using the Law of Mass Action are two key assumptions that should be mentioned: continuity and determinism. With regards to the continuity assumption, it is important to note that the individual genes are often only present in one or two copies per cell. Therefore, there are only one or two regulatory regions to which the regulatory molecules can bind. In addition, the regulatory molecules that bind to these regions are typically produced in low quantities: there may be only a few tens of molecules of a transcription factor in the cell nucleus. This has been shown explicitly in bacterial cells, but there is ample evidence supporting this fact in eukaryotic cells as well (Davidson, 1986; Guptasarma, 1995). The low number of molecules may compromise the notion of continuity.

As for determinism, the rates of some of these reactions are so slow that many minutes may pass before, for instance, the start of mRNA transcription after the necessary molecules are present, or between the start and finish of mRNA creation (Davidson, 1986). This may call into question the notion of the deterministic change presupposed by the use of the differential operator due to the fluctuations in the timing of cellular events. As a consequence, two regulatory systems having the same initial conditions might ultimately settle into different states, a phenomenon strengthened by the small numbers of molecules involved.

There have been some recent experimental results that strongly suggest that cells do in fact behave stochastically. A review can be found in a recent article by the pioneers

of modeling stochastic processes in biology, and they drive home the point that regulatory molecules are present in very low concentrations in cells, with a few hundred being an upper limit, and dozens being a normal phenomenon (McAdams and Arkin, 1999). A study of these systems has shown that the stochastic fluctuations in such a system can produce erratic distributions in protein levels between the same type of cell in a population (McAdams and Arkin, 1997). This is especially true when the molecule under investigation is part of the regulatory mechanism of the cell (Arkin et al., 1998). Most recently, a study in yeast has produced intriguing data concerning the noise in a biological system due to the intrinsic fluctuations (Elowitz et al., 2002).

When the fluctuations in the system are small, it is possible to use a reaction rate equation approach. But when fluctuations are not negligibly small, the reaction rate equations will give results that are at best misleading (showing only the mean behavior), and possibly very wrong if the fluctuations can give rise to important effects. The real problem arises in that it is not always known beforehand whether fluctuations are important. The only way to find out is to use a stochastic simulation: If several stochastic trajectories give results that appear to be identical, then reaction rate equations could indeed have been used. But if the differences in the trajectories were noticeable, then reaction rate equations probably would not have been appropriate. It is possible to forge ahead, and the result is usually a mathematical model that describes the phenomena, but fails to capture the fluctuations present in the system.

Some of the concerns about fluctuations in a system have been around for a long time, if only in theory. With regards to the number of molecules in a cell, this was first mentioned in the English literature by the biochemist J. B. S. Haldane when he

mentioned that critical processes might be carried out by one of a few enzymes per cell (Haldane, 1930). Fifteen years later, this was repeated as a known fact in *Nature* (McIlwain, 1946). More recently there appeared a paper on the question of whether the laws of chemistry apply to living cells (Halling, 1989). It isn't quite as elegant as Purcell's paper on life at low Reynolds numbers (Purcell, 1977), but like this famous talk, the paper points out that it is a very different world inside a cell.

Consequently, the fluctuations in the system may actually be an important part of the system. With these concerns in mind, it seems only natural to investigate an approach that incorporates the small volumes and small number of molecular species (and the inherent fluctuations that are present in a system) and may actually play an important part. These investigations are still relatively new, but in recent years the stochastic simulation algorithm has been used to model phage λ infected *E. coli* cells (Arkin et al., 1998), and calcium wave propagation in rat hepatocytes (Gracheva et al., 2001).

Stochastic Solution

The first mention of using stochastic methods to model chemical reactions appeared in 1940 (Delbrück, 1940; Kramers, 1940). But it wasn't until the early 1950s that it became clear that in small systems the Law of Mass Action breaks down (Rényi, 1954) and even small fluctuations in the number of molecules may be a significant factor in the behavior of the system (Singer, 1953). Soon after, it became evident that some processes in biological cells fell into this category and that a proper mathematical formulation of the chemical reactions in the cells will most likely be based on stochastics (Bartholomay, 1958).

The stochastic approach considers the sets of possible reactions and examines the possible transitions of the system. As an example, consider the following irreversible unimolecular reaction



which is common in radioactive decay processes. In words, the molecule A is converted to B with rate parameter k . The stochastic description of the system is characterized in the following manner. Let $X(t)$ be a random variable that denotes the number of A molecules at time t . Then

- 1) The probability of a transition from $(x + 1)$ molecules to (x) molecules in the interval $(t, t + \Delta t)$ is $k(x + 1)\Delta t + o(\Delta t)$. k is the rate constant and $o(\Delta t)$ takes the usual meaning that $o(\Delta t)/\Delta t \rightarrow 0$ as $\Delta t \rightarrow 0$.
- 2) The probability of a transition from (x) to $(x - j)$, $j > 1$ in the interval $(t, t + \Delta t)$ is $o(\Delta t)$.
- 3) The probability of a transition from (x) to $(x + j)$, $j \geq 1$ in the interval $(t, t + \Delta t)$ is zero.

Denoting the probability of $X(t) = x$ by $P_x(t)$, a balance of the terms yields

$$P_x(t + \Delta t) = k(x + 1)\Delta t P_{x+1}(t) + (1 - kx\Delta t)P_x(t) + o(\Delta t). \quad (2.31)$$

Simplifying and taking the limit $\Delta t \rightarrow 0$ yields the differential-difference equation

$$dP_x(t)/dt = k(x + 1)P_{x+1}(t) - kP_x(t), \quad (2.32)$$

which is also called the chemical master equation for the system.

The solution of the chemical master equation can be thought of as a Markovian random walk in the space of the reacting variables. It measures the probability of finding

the system in a particular state at any given time, and it can be rigorously derived from a microphysical standpoint (Gillespie, 1992). Analytic solutions of master equations are difficult to come by, but in this example it is possible to transform the differential-difference equation into a partial differential equation through the use of the probability generating function

$$F(s,t) = \sum_{x=0}^{\infty} P_x(t) s^x. \quad (2.33)$$

Substituting (2.33) into (2.32) and simplifying leads to

$$\frac{\partial F}{\partial t} = k(1-s) \frac{\partial F}{\partial s}. \quad (2.34)$$

Given the initial condition $F(s,0) = s^{x_0}$ the solution is then

$$F(s,t) = [1 + (s-1)e^{-kt}]^{x_0}. \quad (2.35)$$

Recall that if $X(t)$ is a random variable, then $E[X(t)]$, the expected value, is defined as

$\sum x P_t(x)$ which is, conveniently enough, $\left. \frac{\partial F}{\partial s} \right|_{s=1}$. Computing this value leads to

$$E\{X(t)\} = x_0 e^{-kt}, \quad (2.36)$$

which is the solution of the Mass Action formulation for the system:

$$\frac{dA}{dt} = -kA. \quad (2.37)$$

Thus, the two representations are consistent. However, this is only true in general for unimolecular reactions (McQuarrie, 1967).

Historically, numerical methods were used to construct solutions to the master equations, but the solutions constructed in this manner have some pitfalls. These include

the need to approximate higher-order moments as a product of lower moments, and convergence issues (McQuarrie, 1967). What was needed was a general method that would solve these sorts of problems and this came with the stochastic simulation algorithm.

Stochastic Simulation Algorithm

Given a set of molecular species $\{S_\mu\}_{\mu=1}^N$ and a set of reactions in which they can participate $\{R_\mu\}_{\mu=1}^N$, the Gillespie algorithm, as it has come to be known, is an exact method for numerically computing the time evolution of a chemical system. By exact it is meant that the results are provably equivalent to the chemical master equation, but at no time is it necessary for the master equation to be written down, much less solved.

The fundamental hypothesis of the method is that the reaction parameter c_μ associated with the reaction R_μ can be defined in the following manner:

$c_\mu \delta t \equiv$ the average probability, to the first order in δt , that a particular combination R_μ of reactant molecules will react in the next time interval δt .

In his original work, Gillespie shows that this definition does in fact have a valid physical basis and in fact the reaction parameter c_μ can be easily connected to the traditional reaction rate constant k_μ (Gillespie, 1976).

The method is based on the joint probability density function $P(\tau, \mu)$, defined by

$P(\tau, \mu) d\tau \equiv$ the probability at time t that the next reaction will occur in the differential time interval $(t + \tau, t + \tau + d\tau)$ and will be of type R_μ .

This is a departure from the usual stochastic approach that starts from the probability function $P(X_1, X_2, \dots, X_N; t)$, defined as the probability that at time t there will be X_1 molecules of S_1 , X_2 molecules of S_2 , ..., and X_N molecules of S_N . By using $P(\tau, \mu)$ as the basis of the approach, it is possible to create a tractable method to compute the time evolution of the system. To construct a formula for this quantity, Gillespie starts by defining the quantity h_μ as the number of distinct molecular reactant combinations for the reaction R_μ . This is nothing more than a combinatorial factor and Table 2.1 lists some example values.

Reaction	h_μ	Reaction order
$* \rightarrow S_j$	1	Zeroth
$S_j \rightarrow S_k$	X_j	First
$S_j + S_k \rightarrow S_l$	$X_j \cdot X_k$	Second
$S_j + S_j \rightarrow S_k$	$X_j(X_j - 1)/2$	Second
$S_i + S_j + S_j \rightarrow S_k$	$X_i X_j(X_j - 1)/2$	Third

Table 2.1 Appropriate combinatorial factors for various reactions. In

actuality, everything can be thought of as a zeroth-, first-, or second-order reaction, or a sequential combination of these, and there is no need for the higher-order reactions.

Combining this definition of h_μ with the previous definition for the reaction parameter c_μ , leads to the conclusion that the probability, to the first order in δt , that a R_μ reaction will occur in the next time interval time δt is therefore

$$h_\mu c_\mu \delta t. \quad (2.38)$$

Now $P(\tau, \mu)d\tau$ can be computed as the product of $P_0(\tau)$, the probability that no reaction occurs in the time interval $(t, t + \tau)$, and $h_\mu c_\mu \delta t$, the probability that the specific reaction R_μ occurs in the next time interval $(t + \tau, t + \tau + d\tau)$:

$$P(\tau, \mu)d\tau = P_0(\tau)h_\mu c_\mu d\tau. \quad (2.39)$$

All that is now required is to calculate the term $P_0(\tau)$. To construct an expression for this term, divide the interval $(t, t + \tau)$ into K subintervals, each of length $\varepsilon = \tau/K$. The probability that none of the reactions $\{R_\mu\}_{\mu=1}^M$ occurs in the time interval $(t + j\varepsilon, t + j\varepsilon + 1)$ (for any arbitrary j) is

$$\prod_{i=1}^M [1 - h_i c_i \varepsilon + o(\varepsilon)] = 1 - \sum_{i=1}^M h_i c_i \varepsilon + o(\varepsilon). \quad (2.40)$$

Since there are K subintervals and the probabilities are mutually exclusive,

$$P_0(\tau) = \left[1 - \sum_{i=1}^M h_i c_i \frac{\tau}{K} + o\left(\frac{\tau}{K}\right) \right]^K. \quad (2.41)$$

But as this expression is valid for any K , even infinitely large ones, the expression can also be written as

$$P_0(\tau) = \lim_{K \rightarrow \infty} \left[1 - \left(\sum_{i=1}^M h_i c_i \tau + o(K^{-1}) \right) / K \right]^K. \quad (2.42)$$

However, this is nothing more than one of the limit formulas for the exponential function, and thus

$$P_0(\tau) = \exp\left(-\sum_{i=1}^M h_i c_i \tau\right). \quad (2.43)$$

Therefore, after defining

$$a_\mu \equiv h_\mu \cdot c_\mu, a_o \equiv \sum_{i=1}^M h_i \cdot c_i, \quad (2.44)$$

the result is an expression for $P(\tau, \mu)$:

$$P(\tau, \mu) = a_\mu \exp[-a_o \tau]. \quad (2.45)$$

Implementation

This algorithm can easily be implemented in an efficient modularized form to accommodate quite large reaction sets of considerable complexity.

For an easy implementation, the joint distribution can be broken into two disjoint probabilities using Bayes' rule:

$$P(\tau, \mu) = P(\tau) \cdot P(\mu|\tau). \quad (2.46)$$

But note that the addition property for probabilities can be used to calculate an alternate form for $P(\tau)$:

$$P(\tau) = \sum_{\mu=1}^M P(\tau, \mu), \quad (2.47)$$

and substituting this into (2.45) leads to values for its component parts:

$$P(\tau) = a_o \exp(-a_o \tau), \quad (2.48)$$

$$P(\mu|\tau) = \frac{a_\mu}{a_o}. \quad (2.49)$$

Given these fundamental probability density functions, the following algorithm can be used to carry out the reaction set simulation:

- 1) Initialization

- a. Set values for the c_μ
- b. Set the initial number of the S_μ reactants
- c. Set $t = 0$, and select a value for t_{\max} , the maximum simulation time

2) Loop

- a. Compute $a_\mu \equiv h_\mu \cdot c_\mu, a_o \equiv \sum_{i=1}^M h_i \cdot c_i$
- b. Generate two random numbers r_1 and r_2 from a uniform distribution on $[0,1]$
- c. Compute the next time interval $\tau = \frac{1}{a_o} \ln\left(\frac{1}{r_1}\right)$ (Draw from the probability density function of (2.48))
- d. Select the reaction to be run by computing μ such that $\sum_{v=1}^{\mu-1} a_v < r_2 a_o \leq \sum_{v=1}^{\mu} a_v$
(Draw from the probability density function of (2.49))
- e. Adjust $t = t + \tau$ and update the S_μ values according to the R_μ reaction that just occurred.
- f. If $t > t_{\max}$, then terminate. Otherwise, goto a.

Because the speed of the SSA is linear with respect to the number of reactions, adding new reaction channels will not greatly increase the runtime of the simulation *i.e.*, doubling either the number of reactions or the number of reactant species doubles (approximately) the total runtime of the algorithm. The speed of the SSA depends more on the number of molecules. This is seen by noting that the computation of the next time

interval in (2c) above depends on the reciprocal of a_0 , a term comprised of, among other things, the number of molecules in the simulation. If the reaction set contains at least one second-order reaction, then a_0 will contain at least one product of species population. In this case the speed of the simulation will fall off like the reciprocal of the square of the population. However, the runtime can be reduced by noting that not all of the a_μ values will need to be recalculated after each pass, but only the ones for which S_μ appears as a reactant in the R_μ reaction. An efficient implementation will take advantage of this fact.

Recent improvements to the algorithm, including a method that does not require the probabilities to be updated after every reaction, are helping to keep the runtime in check (Gibson and Bruck, 2000; Gillespie, 2001). As currently implemented, a typical run of the *Hox* simulation presented in Chapter 3 (without the aforementioned speedups) consists of over 23 million events, and takes less than 6 minutes on a computer with a 2GHz Pentium 4 processor.

Two important points should be noted about the SSA: the solution of a system of coupled chemical reactions by this method is entirely equivalent to the solution of the corresponding stochastic master equations (Gillespie, 1976; Gillespie, 1977c; McQuarrie, 1967), and in the limit of large numbers of reactant molecules, the results of this method are entirely equivalent to the solution of the traditional kinetic differential equations derived from the Law of Mass Action (Gillespie, 1977a).

One added benefit of the SSA is the formalism that is forced on the user. Each reaction in the set must be dealt with explicitly, and the connection between the reacting species (and the roles that they play in other reactions) must be clearly specified. The

fact that this algorithm generates its own (nonuniform) time sample should also be noted. Thus, as the simulation proceeds it generates time samples based on the probability density function of (2.43), *i.e.*, simulation time steps are based on draws from an exponential distribution. This of course is one of the reasons why this algorithm is so robust.

Extensions

In order to apply the concepts involved in Gillespie's algorithm to a collection of cells, the original algorithm must be extended to accommodate the introduction of spatial dependencies of the concentration variables. Work has been done which extends the stochastic simulation algorithm to reaction-diffusion processes, and the modification to the method is straightforward. Diffusion is considered to be just another possible chemical event with an associated probability (Stundzia and Lumsden, 1996). As with all the other chemical events, the diffusion is assumed to be intracellular and the basic idea behind this approach is incorporated into the simulation. But one of the important molecules in the simulation is retinoic acid, an intercellular molecule that acts through cell surface receptors, and so the diffusion must be treated in a larger context.

Introducing a spatial context into the SSA is done by creating an interacting cell population represented as a rectangular array of square cells with nearest neighbor only cell-cell interactions. In this model of interacting cells, it is assumed that each cell is running its own internal program of biochemical reactions.

The fact that simulation of any given reaction generates its own "local" simulation time steps poses something of a problem for a model consisting of more than one cell,

each of which is running a reaction simulation independent of all the other cells. This problem arises when an intercellular event must be accounted for, since the internal simulation times of the two partner cells involved will not in general be the same. When implementing such simulations in serial code on a single-processor machine, converting the algorithm from what is essentially a spatial-scanning method to a temporal-scanning method can solve this problem. This is accomplished by first making an initial spatial scan through all of the cells in the array, and inserting the cells into a priority queue that is ordered from shortest to longest local cell time. All succeeding iterations are then based on the temporal order of the cells in the priority queue. In other words, a cell is drawn from the queue, calculations are performed on the reaction set for that cell, and then the cell is placed back on the queue in its new temporal-ordered position. By doing this there is no need to worry about synchronizing reaction simulations between any pair of neighboring cells.

Each reaction that occurs changes the quantity of at least one reactant. When this happens, the combinatorial factors h_μ change and it is necessary to recalculate the a_μ values. This is one of the drawbacks of the approach: if it weren't for having to recalculate the probabilities at every time step, the system is a Markov process with a fixed transition matrix and all standard analysis tools can be brought to bear. In general, only a small number of the a_μ will actually have to be updated and an efficient implementation needs to take advantage of this fact. After the a_μ values are updated, all cells that changed are reordered into their appropriate new position in the priority queue.

Because cells are stored as C-language structures, all of the information required to define the state of any given cell is readily available. The use of a priority queue to order the cells was a unique innovation, and solves the synchronizing problem inherent in a multicellular situation. Not only does this allow an easy mechanism for intercellular signaling, but this methodology can also readily accommodate local inhomogeneities in the molecular populations.

Comparison of the Approaches

The programming language *Mathematica* was used to construct a numerical solution to the original set of differential equations in (2.4) and (2.5). *Mathematica* uses an Adams Predictor-Corrector method for non-stiff differential equations and backward difference formulas (Gear method) for stiff differential equations. It switches between the two methods using heuristics based on the adaptively selected step size. It starts with the non-stiff method, and checks for the advisability of switching methods every 10 or 20 steps. The result is an interpolating function that can be used to construct graphs of the solution for any time interval of interest.

Mathematica was also used to implement the stochastic simulation algorithm for the Michaelis-Menten basic enzyme reaction. This boiled down to a very short piece (less than 25 lines) of code and is included in Appendix D.

Plots of the trajectories of these two methods can be seen in Figure 2.1 below and the reader can easily see the differences between the stochastic and differential equation solutions to the basic enzyme reaction.

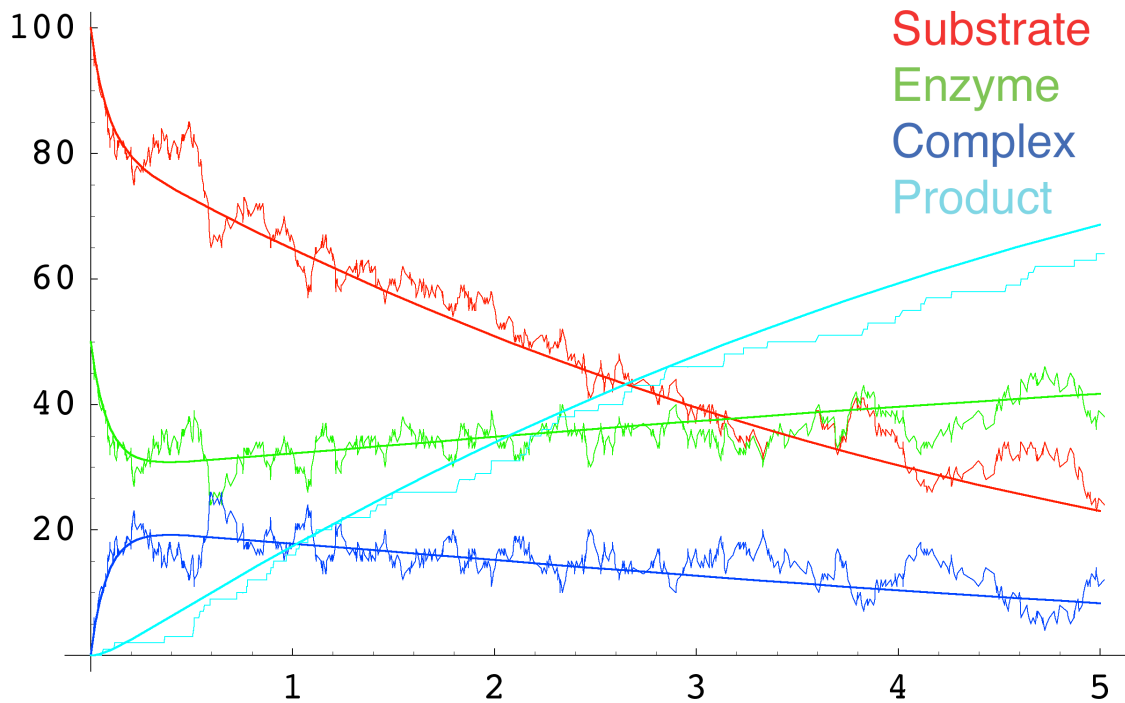


Figure 2.1 Typical solutions of the Michaelis-Menten basic enzyme reaction, low numbers. Both stochastic solution and differential equation solutions to the basic enzyme reaction are shown. The vertical axis is number of molecules and the horizontal axis is time in seconds. Notice that the fluctuations around the differential equations can range up to 50% of the solution when there are low quantities of molecules. The parameters used were $s_0=100$, $e_0=50$, $c_0=p_0=0$, $k_1 = .005$, $k_{-1} = 5.0$, $k_2 = 1.0$.

“On average,” the solutions are the same, but the stochastic approach captures the fluctuations in the system. Notice that there are some marked differences in these solutions. For instance, in the differential equation solution there are always fewer molecules of the complex than there are of the enzyme, but this is not true for the stochastic solution: at about .6 seconds the lines numbers coalesce. Another difference

can be seen in a comparison of the numbers of the substrate and the enzyme. Both the stochastic solution and the differential equation solution meet at about 3.2 seconds, but in the stochastic solution these quantities are very closely matched for the next .5 seconds while the differential equations solution quickly diverge.

But compare Figure 2.1 with Figure 2.2. The rate parameters have not been changed for this figure, only the starting numbers of substrate and enzyme. In this instance the reaction rate method and the stochastic method are in close agreement, both qualitatively and quantitatively.

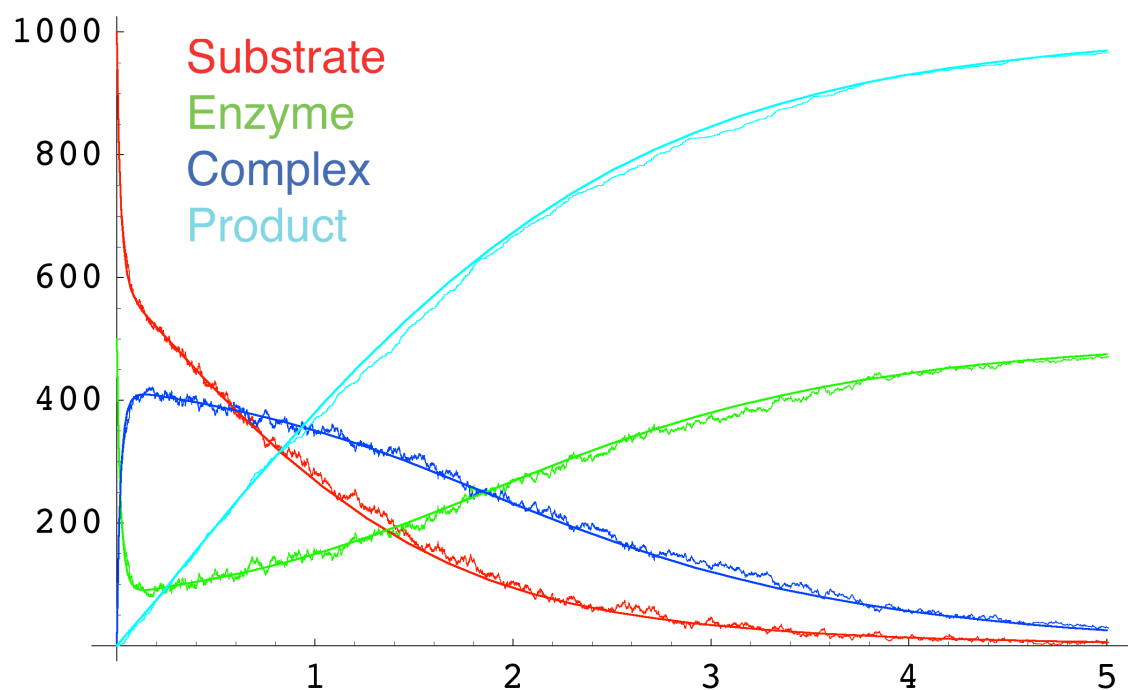


Figure 2.2 Typical solutions of the Michaelis-Menten basic enzyme reaction, high numbers. When there are a large number of molecules, the fluctuations are much less noticeable. The parameters used were $s_0=1000$, $e_0=500$, $c_0=p_0=0$, $k_1 = .005$, $k_{-1} = 5.0$, $k_2 = 1.0$.

In comparing Figures 2.1 and 2.2, it is clear that when the number of molecules is large, the fluctuations might take the appearance of noise. But when there are small numbers of molecules, the fluctuations are may in fact no longer be just noise and may in fact be a significant part of the signal. Whether these fluctuations make a difference in the basic behavior of the system depends on the characteristics of that particular system. In the basic enzyme reaction the fluctuations do not matter, while in the *Notch-Delta* system described below they do. It may also be the case that the system moves between situations in which the fluctuations do and do not matter. Automatically detecting the need for a transition between these situations is part of an ongoing investigation (D. Gillespie, personal communication). However, when it is known that the system contains small numbers of molecules and the network is nonlinear—both of which are true for the *Hox* network—the stochastic approach appears to be a more appropriate method, because both of these situations will magnify any fluctuations that already exist in the system.

Notch-Delta Lateral Inhibition

As previously mentioned, the SSA is an exact method (*i.e.*, the results are provably equivalent to the chemical master equation) for numerically computing the time evolution of a chemical system. It was also proved that in the limit of large numbers of reactant molecules, the results of the SSA method are consistent with the solution of the traditional kinetic differential equations derived from the Law of Mass Action (Gillespie, 1976; Gillespie, 1977c; McQuarrie, 1967). This is not surprising, because the first moment solution to the master equation describes the mean behavior of the system, just as the ODE solution does. But an interesting question concerns the practical connection

of these two methods: in practice, do the two different approaches yield similar results in a system that is sensitive to fluctuations? A related question is what exactly constitutes a large number of molecules.

These questions were explored by modeling lateral inhibition, the process by which a cell adopting a particular fate is able to prevent its neighbors from adopting the same fate. The *Notch-Delta* receptor-ligand pair is found to be involved in lateral inhibition in the cell fate specification in the developing nervous systems (Artavanis-Tsakonas et al., 1995; Chitnis, 1995). A simplified view of this process is shown in Figure 2.3 below.

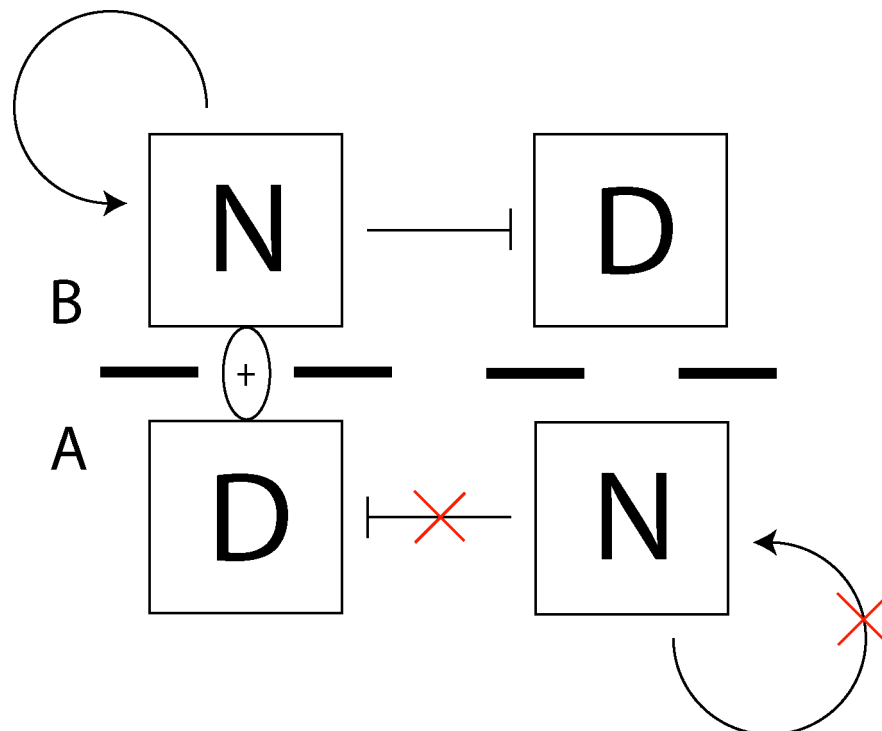


Figure 2.3 Notch-Delta lateral inhibition. When a *Delta* ligand in cell A binds (denoted by the plus inside the oval) to a *Notch* receptor in cell B, the *Notch*

undergoes a modification into an activated form. The activated *Notch* up-regulates *Notch* and down-regulates *Delta* in that cell. On the other hand, the down-regulation of *Delta* in cell B results in fewer *Notch* bindings in cell A. Because of this, activated *Notch* is not formed, and so *Notch* is not up-regulated and *Delta* is not down-regulated in cell A. The collection of events results in cell A becoming *Notch* dominant, and cell B becoming *Delta* dominant.

A study of the *Notch-Delta* lateral inhibition network using ODEs to model the network was undertaken a few years (Collier et al., 1996). The authors of the work examined three situations: a two-cell system, an infinite line, and a two-dimensional grid of cells. The former case was examined using phase plane analysis, while the latter cases were examined numerically using a Runge-Kutta-Merson method. In the two-cell system, they authors proved that if the feedback is sufficiently strong, one cell becomes *Notch* dominant and the other *Delta* dominant. The infinite line case was modeled using periodic boundary conditions, and the results were as expected; alternating *Notch* and *Delta* dominant cells

The two dimensional set of cells was much more interesting. Again there was a regular spatial periodicity to the cells, but they found that the results were very dependant on the boundary conditions. In particular, the default “checkerboard” solutions appeared only when the boundary conditions were compatible with the pattern, but not if the boundary conditions were not compatible with the pattern. This is one of the concerns with the ODE approach: The boundary conditions exert a very strong effect on the

system. Another concern is that the results are not nearly so regular in biological systems and it is known any cell can adopt the default fate (Greenwald, 1998). Finally, the model was heavily non-dimensionalized and caricatured, and the outputs of the model cannot be readily connected to number of molecules or concentrations (N. Monk, personal communication). Therefore, it seemed that a stochastic simulation of the system evolution might be enlightening.

A SSA model was built using the C programming language, and the complete source code can be found in Appendix D and the accompanying CD-ROM. The simulation consisted of 5 types of reactions (creation of *Notch* and *Delta*, decay of *Notch* and *Delta* and binding) and 5 species of molecules (*Notch* and *Delta* Protein, *Notch* and *Delta* mRNA and Activated *Notch*). The investigation was carried out in a 16-by-16 collection of rectangular cells with nearest neighbor communication. The binding between *Notch* and *Delta* required the use of a priority queue to efficiently synchronize the intercellular events. An example of a typical result is seen in Figure 2.4.

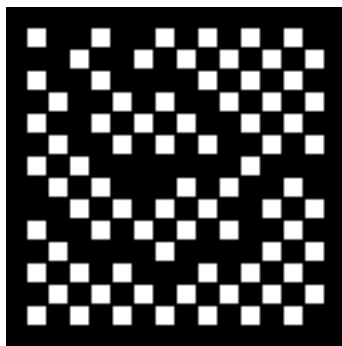


Figure 2.4 *Notch-Delta* lateral inhibition typical results. The white cells are *Notch* dominant, the black cells are *Delta* dominant. Each cell started with 500

molecules each of *Notch* and *Delta* protein, and the simulation was run until equilibrium was reached. Notice that while the cells show general pattern of alternating dominance, there is not strict compliance. This is reflective of the actual pattern of cells as seen in the *Drosophila* (Greenwald, 1998) and so the SSA seems to more accurately predict the observed behavior of cell fate determination than the deterministic approach.

While the stochastic model of *Notch-Delta* lateral inhibition seemed to show results that were consistent with the real cell fate, it was unclear if in the limit of large molecules the stochastic simulation would produce a more regular checkerboard, similar to the deterministic approach. It was also not clear what constitutes a large number of molecules in this case. Therefore, the number of proteins in each cell was increased from the default values of 500 molecules per cell, and results of these simulations are shown in the figures below. Figure 2.5 shows the results when binding does not go beyond the edge of the grid, while Figure 2.6 allows binding to wrap around the edge of the array.

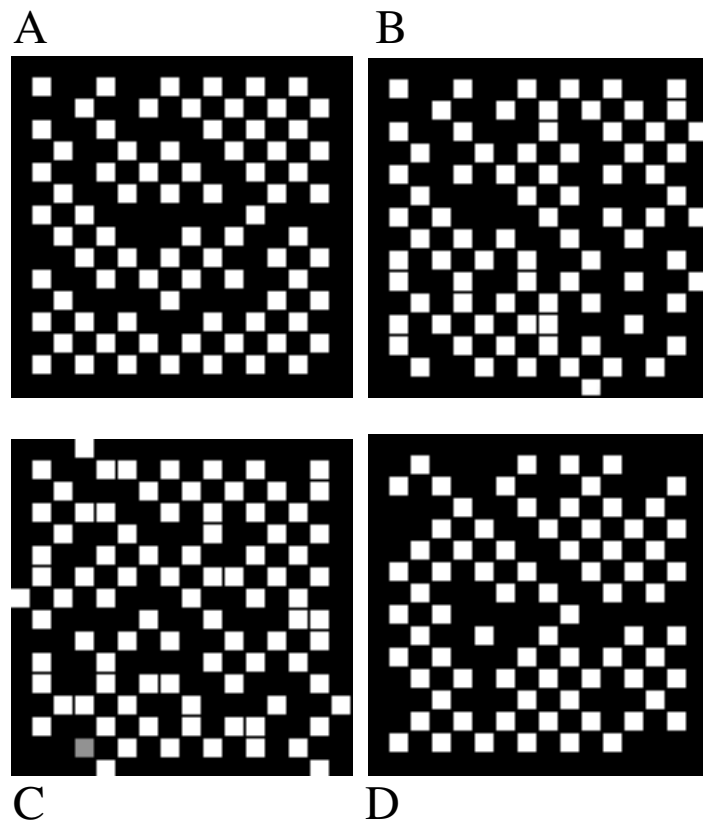


Figure 2.5 *Notch-Delta* larger number results, hard boundary. The white cells are *Notch* dominant, the black cells are *Delta* dominant, and gray cells are ones in which neither is dominant. All input parameters except the starting number of molecules were as in (A) the default case of 500 molecules of *Notch* and *Delta* per cell (B) 1000 molecules per cell (C) 2500 molecules per cell (D) 5000 molecules per cell.

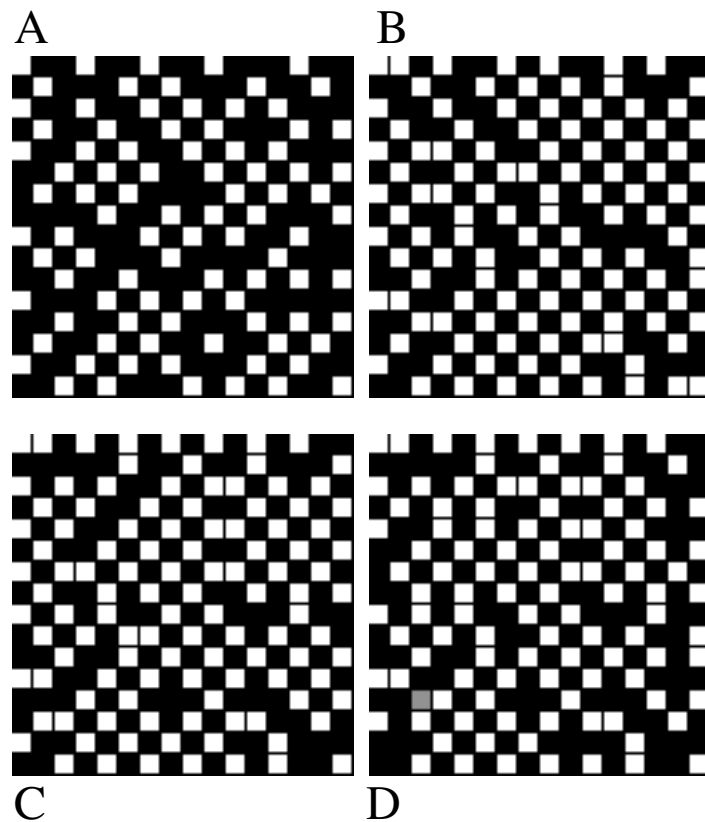


Figure 2.6 Notch-Delta larger number results, wrap around binding. Because the cells only communicate using nearest neighbor connections, the wrap around binding results in a torus of cells. (A) 500 molecules of *Notch* and *Delta* per cell (B) 1000 molecules per cell (C) 2500 molecules per cell (D) 5000 molecules per cell.

Just like in Figure 2.4, all of these results in Figures 2.5 and 2.6 show a general pattern of alternating cell dominance. In Figure 2.5, the number of molecules does not appear to play a role in the regularity of the pattern, but this may be related to the strong role the boundary plays: most of the cells on the edge are *Delta* dominant. Figure 2.6 is

much more interesting. In the larger number cases, the figures appear more regular. To quantify this, the following metric was calculated

$$m = \sum_{\text{Notch cells}} \frac{\# \text{ of adjacent } \textit{Delta} \text{ cells}}{4},$$

and the results are listed in Table 2.2.

	Figure 2.5	Figure 2.6
A	79	91
B	75.25	106.25
C	73	107
D	76	99.5

Table 2.2 Regularity metric values. The regularity metric quantifies the similarity to a perfect checkerboard. The maximum value possible is 128.

The larger numbers do not lead to a more regular pattern for the hard boundary case, while the metric for the torus (Figure 2.6) suggests that the larger numbers of molecules leads to a more regular pattern. For both of these cases however, it should be noted that 5000 molecules per cell is only 1250 of each type per face, and it is not clear that this is yet a “large” number of molecules. Unfortunately, with regards to the *Notch-Delta* simulation 5000 molecules per cell is approaching the practical upper limit of the capabilities of the stochastic simulation. Because one of the reactions is a binding between two different species of molecules, the a_0 value for this reaction contains a

product of terms, and so the speed of the SSA scales quadratically. In addition, the larger number of molecules means that the simulation takes longer to reach equilibrium. So while the results of Figure 2.5A took a little over an hour to generate, Figure 2.5D and Figure 2.6D each took over two days to generate. The deterministic approach is not subject to these sorts of runtime issues, and though the stochastic implementation is exact – even for large number of molecules – this example shows that it is not practical to use in all situations, and deterministic methods will often be a better choice. The stochastic framework appears to be much more at home with small numbers of molecules. Not only does it appear to be on a firmer physical basis than the deterministic approach in this realm (Gillespie, 1976; Gillespie, 1977b; Gillespie, 1992), but the runtime is more likely to be reasonable.

References for Chapter 2

- Arkin, A., Ross, J., and McAdams, H. H. (1998). Stochastic kinetic analysis of developmental pathway bifurcation in phage lambda-infected *Escherichia coli* cells. *Genetics* **149**, 1633-48.
- Artavanis-Tsakonas, S., Matsuno, K., and Fortini, M. E. (1995). Notch Signalling. *Science* **268**, 225-232.
- Bartholomay, A. (1958). Stochastic models for chemical reactions. I. Theory of the unimolecular reaction process. *Bull. Math Biophys.* **20**, 175-190.
- Chitnis, A. B. (1995). The Role of Notch in Lateral Inhibition and Cell Fate Specification. *Mol. Cell Neurosci.* **6**, 311-321.

- Collier, J. R., Monk, N. A., Maini, P. K., and Lewis, J. H. (1996). Pattern formation by lateral inhibition with feedback: a mathematical model of delta-notch intercellular signalling. *J. Theor. Biol.* **183**, 429-46.
- Davidson, E. H. (1986). "Gene activity in early development." Academic Press, Orlando.
- Delbruck, M. (1940). Statistical fluctuations in autocatalytic reactions. *Journal of Chemical Physics* **8**, 120-124.
- Edelstein-Keshet, L. (1998). "Mathematical Models in Biology." McGraw-Hill, Boston.
- Elowitz, M. B., Levine, A., Siggia, E. D., and Swain, P. S. (2002). Stochastic Gene Expression in a Single Cell. *Science* **297**, 1183-1190.
- Gibson, M. A., and Bruck, J. (2000). Efficient Exact Stochastic Simulation of Chemical Systems with Many Species and Many Channels. *J. Phys. Chem. A* **104**, 1876-1889.
- Gillespie, D. T. (1976). A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reactions. *Journal of Computational Physics* **22**, 403.
- Gillespie, D. T. (1977a). Concerning the Validity of the Stochastic Approach to Chemical Kinetics. *Journal of Statistical Physics* **16**, 311-319.
- Gillespie, D. T. (1977b). Concerning the Validity of the Stochastic Approach of Chemical Kinetics. *Journal of Statistical Physics* **16**, 311-319.
- Gillespie, D. T. (1977c). Exact Stochastic Simulation of Coupled Chemical Reactions. *Journal of Physical Chemistry* **81**, 2340-2361.
- Gillespie, D. T. (1992). A rigorous derivation of the chemical master equation. *Physica A* **188**, 404-425.

- Gillespie, D. T. (2001). Approximate accelerated stochastic simulation of chemically reacting systems. *Journal of Chemical Physics* **115**, 1716-1733.
- Gracheva, M. E., Toral, R., and Gunton, J. D. (2001). Stochastic effects in intercellular calcium spiking in hepatocytes. *J. Theor. Biol.* **212**, 111-25.
- Greenwald, I. (1998). LIN-12/Notch signaling: lessons from worms and flies. *Genes Dev* **12**, 1751-62.
- Guptasarma, P. (1995). Does replication-induced transcription regulate synthesis of the myriad low copy number proteins of *E. coli*? *BioEssays* **17**, 987-997.
- Haldane, J. B. S. (1930). "Enzymes." Longmans, Green and Co., London.
- Halling, P. J. (1989). Do the laws of chemistry apply to living cells? *TIBS* **14**, 317-318.
- Kramers, H. A. (1940). Brownian motion in a field of force and the diffusion model of chemical reactions. *Physica* **7**, 284-304.
- McAdams, H. H., and Arkin, A. (1997). Stochastic mechanisms in gene expression. *PNAS* **94**, 814-9.
- McAdams, H. H., and Arkin, A. (1999). It's a noisy business! Genetic regulation at the nanomolar scale. *Trends Genet.* **15**, 65-9.
- McIlwain, H. (1946). The Magnitude of Microbial Reactions Involving Vitamin-like Compounds. *Nature* **158**, 898-902.
- McQuarrie, D. A. (1967). Stochastic Approach to Chemical Kinetics. *Journal of Applied Probability* **4**, 413-478.
- Michaelis, L., and Menten, M. I. (1913). Die Kinetik der Invertinwirkung. *Biochem. Z.*, 333-369.
- Murray, J. D. (1993). "Mathematical Biology." Springer-Verlag, Berlin.

- Purcell, E. M. (1977). Life at Low Reynolds Number. *American Journal of Physics* **45**, 3-11.
- Rényi, A. (1954). Treatment of chemical reactions by means of the theory of stochastic processes (In Hungarian). *Magyar Tud. Akad. Alkalm. Mat. Int. Kozl.* **2**, 93-101.
- Singer, K. (1953). Application of the theory of stochastic processes to the study of irreproducible chemical reactions and nucleation processes. *Journal of the Royal Statistical Society B.* **15**, 92-106.
- Teich, M. (1992). "A documentary history of biochemistry 1770-1940." Associated University Presses, Cranbury, NJ.
- Waage, P., Gulberg, C.M. and Abrash, H.I. (trans). (1986). Studies Concerning Affinity. *Journal of Chemical Education* **63**, 1044-1047.