APPENDIX A

The Continuous Approximation Proof

The probability distribution, $\hat{p}_N(t)$ for the population size, N can be approximated by the solution of the following differential equation:

$$\frac{\partial \hat{p}_N(t)}{\partial t} \approx -\frac{\partial}{\partial N}\left[\left(B(N)-D(N)\right)\hat{p}_N(t)\right]+\frac{1}{2}\frac{\partial^2}{\partial N^2}\left[\left(B(N)+D(N)\right)\hat{p}_N(t)\right] \tag{A.1}$$

Proof: Over the short time interval dt the corresponding change in the population size, dN is a random variable which can, if dt is small enough, only assume the following three values: -1, 0 or 1. It follows that:

$$\begin{aligned} E[dN] &= (1)\times(\Pr[\text{Birth occurs during } dt]) \\ &\quad +(-1)\times(\Pr[\text{Death occurs during } dt]) \\ &\quad +(0)\times(\Pr[\text{No change occurs during } dt]) \\ \Leftrightarrow E[dN] &= [B(N)-D(N)]dt \end{aligned} \tag{A.2}$$

By exactly similar reasoning, it follows that:

$$E\left[(dN)^2\right]=[B(N)+D(N)]dt \tag{A.3}$$

In effecting the transition of N to its comparable continuous representation, we will require that the first two moments of the probability distribution for dN satisfy the above two equations. We make N continuous by moving the 'states' for the birth and death process closer together until they are separated by an amount ε (<<1) – contrast this with the former basic states-separation of one. Accordingly, we assume that dN can assume only the following three values: $-\varepsilon$, 0 and ε. Let:

$$\Pr[dN=\varepsilon]=U(N,\varepsilon)dt,\ \Pr[dN=-\varepsilon]=V(N,\varepsilon)dt \tag{A.4}$$

We then have an equation not dissimilar to the original Kolmogorov equations:

$$\frac{\hat{p}_N(t+dt)-\hat{p}_N(t)}{dt}=U(N-\varepsilon,\varepsilon)\hat{p}_{N-\varepsilon}(t)+V(N+\varepsilon,\varepsilon)\hat{p}_{N+\varepsilon}(t)-[U(N,\varepsilon)+V(N,\varepsilon)]\hat{p}_N(t) \tag{A.5}$$

Performing a Taylor expansion around N on the functions $U(N-\varepsilon,\varepsilon)\hat{p}_{N-\varepsilon}(t)$ and $V(N+\varepsilon,\varepsilon)\hat{p}_{N+\varepsilon}(t)$ we have:

$$\begin{aligned} U(N-\varepsilon,\varepsilon)\hat{p}_{N-\varepsilon}(t) &= U(N,\varepsilon)\hat{p}_N(t)-\varepsilon\frac{\partial}{\partial N}[U(N,\varepsilon)\hat{p}_N(t)]+\frac{1}{2}\varepsilon^2\frac{\partial^2}{\partial N^2}[U(N,\varepsilon)\hat{p}_N(t)]+o(\varepsilon^3) \\ V(N+\varepsilon,\varepsilon)\hat{p}_{N+\varepsilon}(t) &= V(N,\varepsilon)\hat{p}_N(t)-\varepsilon\frac{\partial}{\partial N}[V(N,\varepsilon)\hat{p}_N(t)]+\frac{1}{2}\varepsilon^2\frac{\partial^2}{\partial N^2}[V(N,\varepsilon)\hat{p}_N(t)]+o(\varepsilon^3) \end{aligned} \tag{A.6}$$

Substituting these equations into (A.5), and ignoring terms of third order and higher, we have:

$$\frac{\widehat{p}_N(t+dt)-\widehat{p}_N(t)}{dt}=-\varepsilon\frac{\partial}{\partial N}\left[\left(U(N,\varepsilon)-V(N,\varepsilon)\right)\widehat{p}_N(t)\right]+\frac{1}{2}\varepsilon^2\frac{\partial^2}{\partial N^2}\left[\left(U(N,\varepsilon)+V(N,\varepsilon)\right)\widehat{p}_N(t)\right] \tag{A.7}$$

We saw that, under the continuous representation, the random variable dN takes on the values $-\varepsilon$, 0 and ε. The corresponding expressions for the expectations are:

$$E\left[dN\right]=\varepsilon\left[U(N,\varepsilon)-V(N,\varepsilon)\right]dt \tag{A.8}$$

$$E\left[\left(dN\right)^2\right]=\varepsilon^2\left[U(N,\varepsilon)+V(N,\varepsilon)\right]dt \tag{A.9}$$

We equate these expressions to those given in equations (A.2) and (A.3). After some algebra, we have:

$$U(N,\varepsilon)dt=\frac{1}{2}B(N)dt\left(\frac{1}{\varepsilon}+\frac{1}{\varepsilon^2}\right)-\frac{1}{2}D(N)dt\left(\frac{1}{\varepsilon}-\frac{1}{\varepsilon^2}\right) \tag{A.10}$$

$$V(N,\varepsilon)dt=-\frac{1}{2}B(N)dt\left(\frac{1}{\varepsilon}-\frac{1}{\varepsilon^2}\right)+\frac{1}{2}D(N)dt\left(\frac{1}{\varepsilon}+\frac{1}{\varepsilon^2}\right) \tag{A.11}$$

Substituting the above expressions into equation (A.7) and letting $\varepsilon\to 0$ and $dt\to 0$ in such a way so that dt/ε^2 remains constant, we have:

$$\frac{\partial\widehat{p}_N(t)}{\partial t}\approx-\frac{\partial}{\partial N}\left[\left(B(N)-D(N)\right)\widehat{p}_N(t)\right]+\frac{1}{2}\frac{\partial^2}{\partial N^2}\left[\left(B(N)+D(N)\right)\widehat{p}_N(t)\right]$$

□

APPENDIX B

The Differential Equation for the Moment Generating Function

Bailey (1964) derived the following differential equation for the moment generating function:

$$\frac{\partial M(\theta,t)}{\partial t} = \sum_{j\neq 0}(e^{j\theta}-1)f_j\left(\frac{\partial}{\partial\theta}\right)M(\theta,t) \tag{B.1}$$

where $f_j(N)$ is a transition rate (defined in equation (3.3.7)). The proof for the above equation is shown below:

Proof: Let $\underset{t}{E}$ be the expectation of some function at time t with respect to variations in the population size, $N(t)$ and let $\underset{\Delta t|t}{E}$ be the conditional expectation at the end of the interval Δt for variations in the variable $\Delta N(t)$, given the value of $N(t)$ at time t.

It then follows that:

$$\begin{aligned} M(\theta,t+\Delta t) &= \underset{t+\Delta t}{E}\left[e^{\theta N(t+\Delta t)}\right] \\ &= \underset{t+\Delta t}{E}\left[e^{\theta N(t)+\theta\Delta N(t)}\right] \\ &= \underset{t}{E}\left[e^{\theta N(t)}\ \underset{\Delta t|t}{E}\left[e^{\theta\Delta N(t)}\right]\right] \end{aligned} \tag{B.2}$$

From (B.2) it follows that

$$\begin{aligned} \frac{\partial M}{\partial t} &= \lim_{\Delta t\to 0}\frac{M(\theta,t+\Delta t)-M(\theta,t)}{\Delta t} \\ &= \lim_{\Delta t\to 0}\frac{1}{\Delta t}\left[\underset{t}{E}\left[e^{\theta N(t)}\ \underset{\Delta t|t}{E}\left[e^{\theta\Delta N(t)}\right]\right]-\underset{t}{E}\left[e^{\theta N(t)}\right]\right] \\ &= \underset{t}{E}\left[e^{\theta N(t)}\lim_{\Delta t\to 0}\underset{\Delta t|t}{E}\left[\frac{e^{\theta\Delta N(t)}-1}{\Delta t}\right]\right] \end{aligned} \tag{B.3}$$

If we let $\Psi(\theta,t,X) = \lim_{\Delta t\to 0}\underset{\Delta t|t}{E}\left[\frac{e^{\theta\Delta N(t)}-1}{\Delta t}\right]$, we can express equation (B.3) as:

$$\begin{aligned} \frac{\partial M(\theta,t)}{\partial t} &= \underset{t}{E}\left[e^{\theta N(t)}\Psi(\theta,t,X)\right] \\ &= \Psi\left(\theta,t,\frac{\partial}{\partial\theta}\right)M(\theta,t) \end{aligned} \tag{B.4}$$

Equation (B.4) assumes that the differential and expectation operators are commutable. Note that the $\partial/\partial\theta$ operator acts only on $M(\theta,t)$.

Using the conditional transition probabilities defined in equation (3.3.7), we then have:

$$\begin{aligned}\Psi(\theta,t,N) &= \lim_{\Delta t\to 0} \underset{\Delta t|t}{E}\left[\frac{e^{\theta\Delta N(t)}-1}{\Delta t}\right] \\ &= \lim_{\Delta t\to 0}\frac{\left[1-\Delta t\sum_{j\neq 0} f_j(N)+o(\Delta t)\right]+\sum_{j\neq 0}\left[f_j(N)\Delta t+o(\Delta t)\right]e^{j\theta}-1}{\Delta t} \\ &= \sum_{j\neq 0}\left(e^{j\theta}-1\right)f_j(N) \end{aligned} \qquad \text{(B.5)}$$

Substituting equation (B.5) into equation (B.4), we have:

$$\frac{\partial M(\theta,t)}{\partial t} = \sum_{j\neq 0}(e^{j\theta}-1)f_j\left(\frac{\partial}{\partial\theta}\right)M(\theta,t) \qquad \square$$

APPENDIX C

Fourier Analysis

Any function, $f(t)$ with period λ can be synthesized by a superposition of sinusoidal functions of varying frequencies. The weighting that a sinusoidal function with frequency ω is given (in synthesizing the function $f(t)$) is determined by the amplitude of the sinusoidal variation: the larger the amplitude given to the sinusoidal function whose frequency is ω, the more dominant is that frequency within $f(t)$. The weighting given to the sinusoidal function whose frequency is ω is determined by the corresponding Fourier transform, $\tilde{f}(\omega)$. In general:

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} \tilde{f}(\omega)\, cos(\omega t)\, d\omega = \frac{1}{2\pi}\int_{-\infty}^{\infty} \tilde{f}(\omega)\, e^{i\omega t}\, d\omega \tag{C.1}$$

The Fourier transform, $\tilde{f}(\omega)$ is calculated using the equation:

$$\tilde{f}(\omega) = \int_{-\lambda/2}^{\lambda/2} f(\omega)\, e^{-i\omega t}\, dt \tag{C.2}$$

One can see that equation (C.2) is indeed correct by substituting (C.2) into equation (C.1). We get:

$$RHS = \frac{1}{2\pi}\int_{-\infty}^{\infty}\left[\int_{-\lambda/2}^{\lambda/2} f(t')\, e^{-i\omega t'}\, dt'\right] e^{i\omega t}\, d\omega = \int_{-\lambda/2}^{\lambda/2} f(t')\left[\frac{1}{2\pi}\int_{-\infty}^{\infty} e^{i\omega(t-t')}\, d\omega\right] dt'$$

$$= \int_{-\lambda/2}^{\lambda/2} f(t')\,\delta_{Dirac}(t-t')\, dt' = f(t) \qquad (\text{since } \delta_{Dirac}(t-t') = \frac{1}{2\pi}\int_{-\infty}^{\infty} e^{i\omega(t-t')}\, d\omega)$$

The above workings show that if one combines an infinite number of sinusoidal functions, $\{cos(\omega t) \mid \omega \in (-\infty;\infty)\}$ such that each function has the corresponding weight $\tilde{f}(\omega)\, d\omega$, then the resulting function will be $f(t)$.

Evaluating equation (C.2) at zero, we find:

$$\tilde{f}(0) = \int_{-T/2}^{T/2} f(t)dt = T\langle f\rangle$$

By rearranging the terms, we find:

$$\langle f\rangle = \frac{1}{T}\tilde{f}(0) \tag{C.3}$$

This result is useful for deriving the time average of a function.

Fourier analysis enables one to derive the autocorrelation function, $C(\tau)$ for a stochastic differential equation. In order to work out the autocorrelation function for the random function $f(t)$, one needs to know $f(t)$'s spectral density, $S(\omega)$. The spectral density, $S(\omega)$ for a random function $f(t)$ (where $f(t)$ has zero mean) represents the variability of $f(t)$ in the frequency band ω to $\omega + d\omega$. Formally, $S(\omega)$ is defined as:

$$S(\omega) = \lim_{\lambda\to\infty} \frac{\left|\tilde{f}(\omega)\right|^2}{\lambda} \tag{C.4}$$

Nisbet et al. (1982) showed that the autocovariance function, $C(\tau)$ is given by:

$$C(\tau) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\, e^{i\omega\tau}\, d\omega \tag{C.5}$$

<u>Proof:</u>

$$\begin{aligned}
C(\tau) &= \lim_{\lambda\to\infty} \frac{1}{2\pi\lambda}\int_{-\infty}^{\infty}\left[\int_{-\lambda/2}^{\lambda/2} f(t)e^{+i\omega t}\, dt\right]\left[\int_{-\lambda/2}^{\lambda/2} f(t')e^{-i\omega t'}\, dt'\right]e^{i\omega\tau}\, d\omega \\
&= \lim_{\lambda\to\infty} \frac{1}{2\pi\lambda}\int_{-\lambda/2}^{\lambda/2}\int_{-\lambda/2}^{\lambda/2}\int_{-\infty}^{\infty} f(t)f(t')e^{i\omega(t+\tau-t')}\, d\omega\, dt'\, dt \\
&= \lim_{\lambda\to\infty} \frac{1}{\lambda}\int_{-\lambda/2}^{\lambda/2}\int_{-\lambda/2}^{\lambda/2} f(t)f(t')\left[\frac{1}{2\pi}\int_{-\infty}^{\infty} e^{i\omega(t+\tau-t')}\, d\omega\right] dt'\, dt \\
&= \lim_{\lambda\to\infty} \frac{1}{\lambda}\int_{-\lambda/2}^{\lambda/2}\int_{-\lambda/2}^{\lambda/2} f(t)f(t')\,\delta_{Dirac}(t+\tau-t')\, dt'\, dt \\
&= \lim_{\lambda\to\infty} \frac{1}{\lambda}\int_{-\lambda/2}^{\lambda/2} f(t)f(t+\tau)\, dt
\end{aligned}$$

Because $\langle f(t)\rangle = 0$ (since $f(t)$ has zero mean), we know that the term $\lim_{\lambda\to\infty} \frac{1}{\lambda}\int_{-\lambda/2}^{\lambda/2} f(t)f(t+\tau)\, dt$ is simply the autocorrelation function of $f(t)$. It also immediately follows that:

$$C(0) = {\sigma_{tim}}^2 = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\, d\omega \tag{C.6}$$

APPENDIX D

The Fast Fourier Transform

Consider a function $f(t)$, defined over the range $-\infty < t < \infty$, that is a convolution of two other functions $g(t)$ and $h(t)$ respectively. That is:

$$f(t) = \int_{-\infty}^{\infty} g(\tau)h(t-\tau)\,d\tau \qquad \text{(D.1)}$$

The Fourier transform of $f(t)$ is:

$$\begin{aligned}
\tilde{f}(\omega) &= \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty} g(\tau)h(t-\tau)\,d\tau\right]e^{-i\omega t}\,dt \\
&= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g(\tau)e^{-i\omega t}h(t-\tau)e^{-i\omega(t-\tau)}\,d\tau\,dt \\
&= \int_{-\infty}^{\infty} g(\tau)e^{-i\omega t}\left[\int_{-\infty}^{\infty} h(z)e^{-i\omega z}\,dz\right]d\tau \qquad \text{where } z = t-\tau
\end{aligned} \qquad \text{(D.2)}$$

The term $\int_{-\infty}^{\infty} h(z)e^{-i\omega z}\,dz$ is simply the Fourier transform of $h(t)$ (denoted by $\tilde{h}(\omega)$). We can thus take the term out of the integral in (D.2) to obtain:

$$\tilde{f}(\omega) = \tilde{h}(\omega)\int_{-\infty}^{\infty} g(\tau)e^{-i\omega\tau}\,d\tau = \tilde{g}(\omega)\tilde{h}(\omega) \qquad \text{(D.3)}$$

Substituting equation (D.3) within equation (C.1), we have:

$$f(t) = \int_{-\infty}^{\infty} \tilde{g}(\omega)\tilde{h}(\omega)e^{i\omega t}\,d\omega \qquad \text{(D.4)}$$

Instead of calculating the convolution directly from (D.1), it is often easier to evaluate a convolution ($= f(t)$) using equation (D.4).

APPENDIX E

The Differential Equation for the Cumulant Generating Function

Bailey derived a differential equation for the moment generating function, $M(\theta,t)$ that was based on the population transition rates. From the differential equation for $M(\theta,t)$, one can quickly derive a corresponding differential equation for the cumulant generating function, $K(\theta,t)$. The differential equation for $M(\theta,t)$ can be obtained by applying the result derived by Bailey (1964) to a given set of population transition rates. For every set of transition rates considered within this report, the corresponding differential equation for $K(\theta,t)$ was shown, but not derived.

We now consider how, given the population transition rates, one would derive the corresponding differential equation for $K(\theta,t)$. Suppose the transition rates have the following form:

$$\begin{aligned} &B(N) = f_1(N) = a_1 N - b_1 N^2 \qquad \text{for } N \le \frac{a_1}{b_1} \\ &D(N) = f_{-1}(N) = a_2 N + b_2 N^2 \\ &\text{where } a_i, b_i > 0;\ i = 1, 2 \text{ and } B(0) = D(0) = 0 \end{aligned} \tag{E.1}$$

By applying the result by Bailey (1964), we find:

$$\frac{\partial M}{\partial t} = \left[(e^{\theta}-1)a_1 + (e^{-\theta}-1)a_2\right]\frac{\partial M}{\partial \theta} + \left[(e^{\theta}-1)(-b_1) + (e^{-\theta}-1)b_2\right]\frac{\partial^2 M}{\partial \theta^2} \tag{E.2}$$

If we divide both sides of equation (E.2) by $M(\theta,t)$ we obtain:

$$\frac{1}{M}\frac{\partial M}{\partial t} = \left[\left(e^{\theta}-1\right)a_1 + \left(e^{-\theta}-1\right)a_2\right]\frac{1}{M}\frac{\partial M}{\partial \theta} + \left[\left(e^{\theta}-1\right)\left(-b_1\right) + \left(e^{-\theta}-1\right)b_2\right]\frac{1}{M}\frac{\partial^2 M}{\partial \theta^2} \tag{E.3}$$

Now $d\log M(\theta,t) = \left(1/M(\theta,t)\right)dM(\theta,t)$. It follows:

$$\frac{\partial K}{\partial t} = \left[\left(e^{\theta}-1\right)a_1 + \left(e^{-\theta}-1\right)a_2\right]\frac{\partial K}{\partial \theta} + \left[\left(e^{\theta}-1\right)\left(-b_1\right) + \left(e^{-\theta}-1\right)b_2\right]\frac{1}{M}\frac{\partial^2 M}{\partial \theta^2} \tag{E.4}$$

Now:

$$\begin{aligned} \frac{1}{M}\frac{\partial^2 M}{\partial \theta^2} &= \frac{1}{M}\frac{\partial}{\partial \theta}\left[\frac{\partial M}{\partial \theta}\right] = \frac{1}{M}\frac{\partial}{\partial \theta}\left[M\frac{\partial K}{\partial \theta}\right] \\ &= \frac{1}{M}\left[\frac{\partial M}{\partial \theta}\frac{\partial K}{\partial \theta} + M\frac{\partial^2 K}{\partial \theta^2}\right] \\ &= \left(\frac{\partial K}{\partial \theta}\right)^2 + \frac{\partial^2 K}{\partial \theta^2} \end{aligned} \tag{E.5}$$

Substituting Equation (E.5) in Equation (E.4), we have:

$$\frac{\partial K}{\partial t} = \left[\left(e^{\theta}-1\right)a_1 + \left(e^{-\theta}-1\right)a_2\right]\frac{\partial K}{\partial \theta} + \left[\left(e^{\theta}-1\right)\left(-b_1\right) + \left(e^{-\theta}-1\right)b_2\right]\left[\left(\frac{\partial K}{\partial \theta}\right)^2 + \frac{\partial^2 K}{\partial \theta^2}\right] \tag{E.6}$$

APPENDIX F

The linear approximation to the population time average under sinusoidal environmental variations

Suppose the environemental condition, $\phi(t)$ is governed by the following relationship:

$$\phi(t) = b\cos\omega t \tag{F.1}$$

Also suppose the birth and death rates have the following form:

$$B(N) = rN; \qquad D(N) = \frac{rN^2}{K}\left(1+\phi(t)\right) \tag{F.2}$$

The equilibrium population size when $\phi(t) = 0$ is K whilst the long-term environmental average, ϕ^* is zero. That is:

$$N^* = K; \quad \phi^* = 0 \tag{F.3}$$

Using (F.3), it can thus be shown that:

$$\begin{aligned}\alpha &= \left[\frac{\partial B}{\partial \phi} - \frac{\partial D}{\partial \phi}\right]_{N=N^*,\phi=\phi^*} = -\frac{\partial}{\partial \phi}\left[\frac{rN^2}{K}(1+\phi(t))\right]_{N=N^*,\phi=\phi^*} \\ &= -\left[\frac{rN^2}{K}\right]_{N=N^*,\phi=\phi^*} = -rK\end{aligned} \tag{F.4}$$

Also:

$$\begin{aligned}\lambda &= \left[\frac{\partial B}{\partial N} - \frac{\partial D}{\partial N}\right]_{N=N^*,\phi=\phi^*} = \frac{\partial}{\partial N}\left[rN - \frac{rN^2}{K}(1+\phi(t))\right]_{N=N^*,\phi=\phi^*} \\ &= \left[r - \frac{2rN}{K}(1+\phi(t))\right]_{N=N^*,\phi=\phi^*} = -r\end{aligned} \tag{F.5}$$

By substituting equations (F.4) and (F.5) into equation (5.2.31), we obtain:

$$\langle n(t)\rangle = -\frac{\alpha\tilde{f}(0)}{\lambda T} = -\frac{rK\tilde{f}(0)}{(-r)T} = -\frac{K\tilde{f}(0)}{T} \tag{F.6}$$

Upon substituting equation (5.2.33) into equation (F.6), we get:

$$\langle n(t)\rangle = -\frac{K}{T}\left(bT\right) = -Kb \tag{F.7}$$

We thus have:

$$\begin{aligned}\langle N(t)\rangle &= \left\langle N^* + n(t)\right\rangle = N^* + \langle n(t)\rangle \\ &= K - Kb = K\left(1-b\right) \qquad \text{as required}\end{aligned} \tag{F.8}$$

We have thus proved equation (5.2.36). The derivations of equations (5.2.37) and (5.2.38) are similar.

APPENDIX G

Simulation Programs

The programs used to simulate the various population models are given below together with a brief description of the program. Note that all the programs were macros written within Microsoft Excel.

Program 1

Aim: To simulate one trajectory of population governed by the Kolmogorov equations. The transition rates are assumed to be of the form given in equation (2.1.1).

```
Sub Main()
   Dim a1, a2, b1, b2, s1, s2, timediff, time, dice As Double
   Dim i, N, num As Integer
   a1 = 0.3
   b1 = 0.015
   s1 = 1
   a2 = 0.02
   b2 = 0.001
   s2 = 1
   N = Application.InputBox(Prompt:="Enter the starting population number:", Type:=1)
   time = 100
   timediff = 0.01
   num = time / timediff
   Worksheets("Simulation").Cells(1, 1) = "Time"
   Worksheets("Simulation").Cells(1, 2) = "Popn Size"
   For i = 1 To num
      dice = Rnd
      Worksheets("Simulation").Cells(i + 1, 3) = (((a1 * N) - (b1 * (N ^ (s1 + 1)))) * timediff)
      Worksheets("Simulation").Cells(i + 1, 4) = ((((a1 * N) - (b1 * (N ^ (s1 + 1)))) + ((a2 * N) + (b2 * (N ^ (s2 + 1))))) * timediff)
      Worksheets("Simulation").Cells(i + 1, 5) = dice
      If dice < (((a1 * N) - (b1 * (N ^ (s1 + 1)))) * timediff) Then
         If N < 20 Then
            N = N + 1
         End If
      Else
         If dice < ((((a1 * N) - (b1 * (N ^ (s1 + 1)))) + ((a2 * N) + (b2 * (N ^ (s2 + 1))))) * timediff) Then
            N = N - 1
         End If
      End If
      Worksheets("Simulation").Cells(i + 1, 1) = i * timediff
      Worksheets("Simulation").Cells(i + 1, 2) = N
   Next i
End Sub
```

Program 2

Aim: To simulate the probability distribution of a population governed by the Kolmogorov equations and whose transition rates are of the form (2.1.1).

```
Sub Main()
   Dim a1, a2, b1, b2, s1, s2, timediff, time, dice As Double
   Dim h, i, N, Ninit, num, numS As Long
   a1 = 0.3
   b1 = 0.015
   s1 = 1
   a2 = 0.02
   b2 = 0.001
   s2 = 1
   Dim freqtable(20) As Long
   Ninit = Application.InputBox(Prompt:="Enter the starting population number:", Type:=1)
   numS = Application.InputBox(Prompt:="How many simulations do you want?", Type:=1)
   time = 10
   timediff = 0.01
   num = time / timediff
   Worksheets("ProbDist").Cells(1, 1) = "Popn size"
   Worksheets("ProbDist").Cells(1, 2) = "p(N,1)"
   Worksheets("ProbDist").Cells(1, 3) = "freq(N,1)"
   For i = 0 To 20
      freqtable(i) = 0
   Next i
   For h = 1 To numS
      N = Ninit
      For i = 1 To num
         dice = Rnd
         If dice < (((a1 * N) - (b1 * (N ^ (s1 + 1)))) * timediff) Then
            N = N + 1
         Else
            If dice < ((((a1 * N) - (b1 * (N ^ (s1 + 1)))) + ((a2 * N) + (b2 * (N ^ (s2 + 1))))) * timediff) Then
               N = N - 1
            End If
         End If
      Next i
      freqtable(N) = freqtable(N) + 1
   Next h
   For i = 0 To 20
      Worksheets("ProbDist").Cells(i + 2, 1) = i
      Worksheets("ProbDist").Cells(i + 2, 2) = freqtable(i) / numS
      Worksheets("ProbDist").Cells(i + 2, 3) = freqtable(i)
   Next i
End Sub
```

Program 3

Aim: To simulate a trajectory of a population that was governed by the stochastic differential equation and whose transition rates have a form given by (2.1.1).

```
Sub Main()
    Dim a1, a2, b1, b2, s1, s2, timediff, time, dice As Double
    Dim i, N, num As Long
    a1 = 0.3
    b1 = 0.015
    s1 = 1
    a2 = 0.02
    b2 = 0.001
    s2 = 1
    N = Application.InputBox(Prompt:="Enter the starting population number:", Type:=1)
    time = 100
    timediff = 0.01
    num = time / timediff
    Worksheets("Simulation").Cells(1, 1) = "Time"
    Worksheets("Simulation").Cells(1, 2) = "Popn Size"
    For i = 1 To (num + 1)
        dice = Application.WorksheetFunction.NormInv(Rnd, 0, (timediff ^ 0.5))
        Worksheets("Simulation").Cells(i + 1, 1) = ((i - 1) / 100)
        Worksheets("Simulation").Cells(i + 1, 2) = N
        N = N + ((((a1 * N) - (b1 * (N ^ (s1 + 1)))) - ((a2 * N) + (b2 * (N ^ (s2 + 1))))) * timediff) + (((((a1 * N) - (b1 * (N ^ (s1 + 1)))) + ((a2 * N) + (b2 * (N ^ (s2 + 1))))) ^ 0.5) * dice)
    Next i
End Sub
```

Program 4

Aim: To simulate the probability distribution of a population that is governed by the stochastic differential equations whose transition rates have a form given by (2.1.1).

```
Sub Main()
   Dim a1, a2, b1, b2, s1, s2, timediff, time, dice As Double
   Dim i, j, N, Ninit, num, numS As Long
   a1 = 0.3
   b1 = 0.015
   s1 = 1
   a2 = 0.02
   b2 = 0.001
   s2 = 1
   Dim freqtable(30) As Long
   Ninit = Application.InputBox(Prompt:="Enter the starting population number:", Type:=1)
   numS = Application.InputBox(Prompt:="How many simulations do you want?", Type:=1)
   time = 10
   timediff = 0.01
   num = time / timediff
   Worksheets("ProbDist").Cells(1, 1) = "Popn size"
   Worksheets("ProbDist").Cells(1, 2) = "p(N,1)"
   Worksheets("ProbDist").Cells(1, 3) = "freq(N,1)"
   For i = 0 To 30
      freqtable(i) = 0
   Next i
   For i = 1 To numS
      N = Ninit
      For j = 1 To num
         dice = Application.WorksheetFunction.NormInv(Rnd, 0, (timediff ^ 0.5))
         N = N + ((((a1 * N) - (b1 * (N ^ (s1 + 1)))) - ((a2 * N) + (b2 * (N ^ (s2 + 1))))) * timediff) + (((((a1 * N) - (b1 * (N ^ (s1 + 1)))) + ((a2 * N) + (b2 * (N ^ (s2 + 1))))) ^ 0.5) * dice)
      Next j
      freqtable(Round(N)) = freqtable(Round(N)) + 1
      Worksheets("ProbDist").Cells(Round(N) + 2, 3) = freqtable(Round(N))
      Worksheets("ProbDist").Cells(3, 5) = i
   Next i
   For i = 0 To 30
      Worksheets("ProbDist").Cells(i + 2, 1) = i
      Worksheets("ProbDist").Cells(i + 2, 2) = freqtable(i) / numS
   Next i
End Sub
```

Program 5

Aim: To simulate a population trajectory of a population governed by the branching processes discussed in Chapter 4.

```
Sub Main()
   Dim pred, i, j, N As Integer
   Dim p0, p1, p2, dice As Double
   p0 = 0.2
   p1 = 0.4
   p2 = 0.4
   N = 1
   pred = Application.InputBox(Prompt:="How far into the future?", Type:=1)
   Worksheets("Simulation").Cells(1, 2) = "Time"
   Worksheets("Simulation").Cells(2, 2) = 0
   Worksheets("Simulation").Cells(1, 3) = "Popn Num"
   Worksheets("Simulation").Cells(2, 3) = N
   For i = 1 To pred
      For j = 1 To N
         dice = Rnd
         If (dice < p0) And (Not (N = 0)) Then
            N = N - 1
         ElseIf (dice > (p0 + p1)) And (Not (N = 0)) Then
            N = N + 1
         End If
      Next j
      Worksheets("Simulation").Cells(i + 2, 2) = i
      Worksheets("Simulation").Cells(i + 2, 3) = N
   Next i
End Sub
```

Program 6

Aim: To simulate the probability distribution of a population governed by the Branching processes discussed in Chapter 4.

```
Sub Main()
   Dim pred, i, j, N As Integer
   Dim p0, p1, p2, dice As Double
   Dim h, NumS As Long
   Dim freqtable(1001) As Long
   p0 = 0.2
   p1 = 0.4
   p2 = 0.4
   N = 1
   pred = Application.InputBox(Prompt:="How far into the future?", Type:=1)
   NumS = Application.InputBox(Prompt:="Enter the number of Simulations needed:", Type:=1)
   Worksheets("ProbDist").Cells(1, 2) = "N"
   Worksheets("ProbDist").Cells(1, 3) = "p(N)"
   For i = 0 To 1000
      freqtable(i) = 0
      Worksheets("ProbDist").Cells(i + 2, 3) = 0
      Worksheets("ProbDist").Cells(i + 2, 2) = i
   Next i
   For h = 1 To NumS
      For i = 1 To pred
         For j = 1 To N
            dice = Rnd
            If (dice < p0) And (Not (N = 0)) Then
               N = N - 1
            ElseIf (dice > (p0 + p1)) And (Not (N = 0)) Then
               N = N + 1
            End If
         Next j
      Next i
      freqtable(N) = freqtable(N) + 1
      Worksheets("ProbDist").Cells(N + 2, 3) = freqtable(N)
      Worksheets("ProbDist").Cells(1, 1) = h
      N = 1
   Next h
End Sub
```

Program 7

Aim: To simulate a population trajectory of a population subject to sinusoidal environmental fluctuations and that is governed by the Kolmogorov equations.

```
Sub Main()
   Dim b, w, r, K, timediff, time, dice As Double
   Dim N As Integer
   Dim i, num As Long
   b = 0.1
   r = 0.2
   K = 20
   w = 0.12566
   N = Application.InputBox(Prompt:="Enter the starting population number:", Type:=1)
   time = 600
   timediff = 0.01
   num = time / timediff
   Worksheets("Simulation_BD").Cells(1, 1) = "Time"
   Worksheets("Simulation_BD").Cells(1, 2) = "Popn Size"
   For i = 1 To num
      dice = Rnd
      If dice < ((r * N) * timediff) Then
         N = N + 1
      Else
         If dice < (((r * N) + (((r * N * N) / K) * (1 + (b * Cos(w * (i - 1) * timediff))))) * timediff) Then
            N = N - 1
         End If
      End If
      If ((i Mod 10) = 0) Then
         Worksheets("Simulation_BD").Cells((i / 10) + 1, 3) = i * timediff
         Worksheets("Simulation_BD").Cells((i / 10) + 1, 4) = N
      End If
      Worksheets("Simulation_BD").Cells(i + 1, 1) = i * timediff
      Worksheets("Simulation_BD").Cells(i + 1, 2) = N
   Next i
End Sub
```

Program 8

Aim: To simulate a population trajectory of a population subject to a randomly varying environment that is itself governed by the Ornstein-Uhlenbeck process. The transition rates are given by (5.4.1).

```
Sub Main()
    Dim a1, a2, b1, b2, c1, c2, b, EQ, var, Q, timediff, time, diceN, diceQ As Double
    Dim i, N, num As Integer
    a1 = 0.38
    a2 = 0.08
    b1 = 0.004
    b2 = 0.016
    c1 = 0.02
    c2 = 0.03
    b = 0.5
    EQ = 0
    var = 9
    N = Application.InputBox(Prompt:="Enter the starting population number:", Type:=1)
    Q = Application.InputBox(Prompt:="Enter the starting environmental condition:", Type:=1)
    time = 100
    timediff = 0.01
    num = time / timediff
    Worksheets("Simulation").Cells(1, 1) = "Time"
    Worksheets("Simulation").Cells(1, 2) = "Environment"
    Worksheets("Simulation").Cells(1, 3) = "Popn Size"
    For i = 1 To num
        diceN = Rnd
        diceQ = Rnd
        Worksheets("Simulation").Cells(i + 1, 5) = (((a1 * N) - (b1 * N * N) + (c1 * N * Q)) * timediff)
        Worksheets("Simulation").Cells(i + 1, 6) = (((a1 * N) - (b1 * N * N) + (c1 * N * Q) + (a2 * N) + (b2 * N * N) - (c2 * N * Q)) * timediff)
        If diceN < (((a1 * N) - (b1 * N * N) + (c1 * N * Q)) * timediff) Then
            N = N + 1
        Else
            If diceN < (((a1 * N) - (b1 * N * N) + (c1 * N * Q) + (a2 * N) + (b2 * N * N) - (c2 * N * Q)) * timediff) Then
                N = N - 1
            End If
        End If
        Q = Q + Application.WorksheetFunction.NormInv(diceQ, (b * (EQ - Q)) * timediff, ((var * timediff) ^ 0.5))
        Worksheets("Simulation").Cells(i + 1, 1) = i * timediff
        Worksheets("Simulation").Cells(i + 1, 2) = Q
        Worksheets("Simulation").Cells(i + 1, 3) = N
    Next i
End Sub
```

Program 9

Aim: To simulate the probability distributions for the population and the environment when the population is subject to a randomly varying environment that is itself governed by the Ornstein-Uhlenbeck process. The transition rates are given by (5.4.1).

```
Sub Main()
   Dim a1, a2, b1, b2, c1, c2, b, EQ, var, Q, Qinit, timediff, time, diceN, diceQ As Double
   Dim i, j, N, Ninit, num, numS As Integer
   a1 = 0.38
   a2 = 0.08
   b1 = 0.004
   b2 = 0.016
   c1 = 0.02
   c2 = 0.03
   b = 0.5
   EQ = 0
   var = 9
   Dim Nfreqtable(80) As Integer
   Dim Qfreqtable(40) As Integer
   Ninit = Application.InputBox(Prompt:="Enter the starting population number:", Type:=1)
   Qinit = Application.InputBox(Prompt:="Enter the starting environmental condition:", Type:=1)
   numS = Application.InputBox(Prompt:="How many simulations do you want?", Type:=1)
   time = 5
   timediff = 0.01
   num = time / timediff
   Worksheets("Distributions").Cells(1, 1) = "Popn size"
   Worksheets("Distributions").Cells(1, 2) = "p(N,1)"
   Worksheets("Distributions").Cells(1, 3) = "freq(N,1)"
   Worksheets("Distributions").Cells(1, 5) = "Environment"
   Worksheets("Distributions").Cells(1, 6) = "p(N,1)"
   Worksheets("Distributions").Cells(1, 7) = "freq(N,1)"
   For i = 0 To 80
      Worksheets("Distributions").Cells(i + 2, 1) = i
      Nfreqtable(i) = 0
   Next i
   For i = 0 To 40
      Qfreqtable(i) = 0
      Worksheets("Distributions").Cells(i + 2, 5) = i - 20
   Next i
   For i = 1 To numS
      N = Ninit
      Q = Qinit
      For j = 1 To num
         diceN = Rnd
         diceQ = Rnd
         If diceN < (((a1 * N) - (b1 * N * N) + (c1 * N * Q)) * timediff) Then
            N = N + 1
         Else
            If diceN < (((a1 * N) - (b1 * N * N) + (c1 * N * Q) + (a2 * N) + (b2 * N * N) - (c2 * N * Q)) * timediff) Then
               N = N - 1
            End If
         End If
         Q = Q + Application.WorksheetFunction.NormInv(diceQ, (b * (EQ - Q)) * timediff, ((var * timediff) ^ 0.5))
      Next j
      Nfreqtable(N) = Nfreqtable(N) + 1
      Qfreqtable(Round(Q, 0) + 20) = Qfreqtable(Round(Q, 0) + 20) + 1
      Worksheets("Distributions").Cells(N + 2, 3) = Nfreqtable(N)
      Worksheets("Distributions").Cells(Round(Q, 0) + 22, 7) = Qfreqtable(Round(Q, 0) + 20)
   Next i
End Sub
```

Program 10

Aim: To simulate a population and environmental trajectory of an ecological system. The transition rates are given by (6.3.4) and (6.3.5).

```
Sub Main()
   Dim a1, a2, b1, b2, c1, c2, d1, d2, g1, g2, diceN, diceQ, lam, timediff, IncN, DecN, IncQ, DecQ As Double
   Dim i, j, num, numD As Long
   Dim N, Q, disas, time As Integer
   a1 = 0.05
   a2 = 0.11
   b1 = 0.002
   b2 = 0.0006
   c1 = 0.007
   c2 = 0.003
   d1 = 0.1
   d2 = 0.026
   g1 = 0.0008
   g2 = 0.0003
   Q = 15
   N = 22
   disas = 5
   lam = 0.01
   timediff = 0.01
   time = 300
   num = time / timediff
   numD = 0
   Worksheets("Envir_Traj").Cells(1, 1) = "Time"
   Worksheets("Envir_Traj").Cells(1, 2) = "Q"
   Worksheets("Envir_Traj").Cells(1, 3) = "N"
   Worksheets("Envir_Traj").Cells(2, 1) = 0
   Worksheets("Envir_Traj").Cells(2, 2) = Q
   Worksheets("Envir_Traj").Cells(2, 3) = N
   For i = 1 To num
      diceN = Rnd
      diceQ = Rnd
      IncN = ((c1 * N * Q) - (a1 * N))
      DecN = ((-1 * c2 * N * Q) + (a2 * N))
      IncQ = ((d1 * Q) - (g1 * Q * Q) - (b1 * N * Q))
      DecQ = ((d2 * Q) + (g2 * Q * Q) + (b2 * N * Q))
      If diceN < (IncN * timediff) Then
         N = N + 1
      ElseIf diceN < ((IncN * timediff) + (DecN * timediff)) Then
         N = N - 1
      End If
      If diceQ < (IncQ * timediff) Then
         Q = Q + 1
      ElseIf diceQ < ((IncQ * timediff) + (DecQ * timediff)) Then
         If Q > 1 Then
            Q = Q - 1
         End If
      ElseIf diceQ < ((IncQ * timediff) + (DecQ * timediff) + (lam * timediff)) Then
         Q = Q - disas
         numD = numD + 1
         If Q < 1 Then
            Q = 1
         End If
         Worksheets("Envir_Traj").Cells(numD + 2, 4) = i * timediff
      End If
      Worksheets("Envir_Traj").Cells(i + 2, 5) = (IncN * timediff)
      Worksheets("Envir_Traj").Cells(i + 2, 6) = ((IncN * timediff) + (DecN * timediff))
      Worksheets("Envir_Traj").Cells(i + 2, 8) = (IncQ * timediff)
      Worksheets("Envir_Traj").Cells(i + 2, 9) = ((IncQ * timediff) + (DecQ * timediff))
      Worksheets("Envir_Traj").Cells(i + 2, 10) = ((IncQ * timediff) + (DecQ * timediff) + (lam * timediff))
      Worksheets("Envir_Traj").Cells(i + 2, 1) = i * 0.01
      Worksheets("Envir_Traj").Cells(i + 2, 2) = Q
      Worksheets("Envir_Traj").Cells(i + 2, 3) = N
   Next i
   Worksheets("Envir_Traj").Cells(1, 5) = numD
End Sub
```

Program 11

Aim: To simulate the population and environmental probability distribution of a population whose transition rates are given by (6.3.4) and (6.3.5).

```
Sub Main()
   Dim a1, a2, b1, b2, c1, c2, d1, d2, g1, g2, diceN, diceQ, lam, timediff, tdis, IncN, DecN, IncQ, DecQ As Double
   Dim i, j, num, baff As Long
   Dim N, Q, disas, time, numD, numS As Integer
   a1 = 1.2
   a2 = 8.3
   b1 = 0.05
   b2 = 0.05
   c1 = 0.14
   c2 = 0.02
   d1 = 41
   d2 = 30
   g1 = 0.02
   g2 = 0.05
   Q = 61
   N = 60
   disas = 40
   lam = 1
   numD = 0
   timediff = 0.00002
   tdis = 0.01        'tdis is how often Q and N should be displayed in worksheet
   time = 5
   num = time / timediff
   baff = tdis / timediff
   numS = 25000
   Dim freqtableN(1000) As Integer
   Dim freqtableQ(1000) As Integer
   Worksheets("ProbDists").Cells(1, 1) = "Popn size"
   Worksheets("ProbDists").Cells(1, 2) = "p(N,1)"
   Worksheets("ProbDists").Cells(1, 3) = "freq(N,1)"
   Worksheets("ProbDists").Cells(1, 5) = "Env size"
   Worksheets("ProbDists").Cells(1, 6) = "p(Q,1)"
   Worksheets("ProbDists").Cells(1, 7) = "freq(Q,1)"
   Worksheets("ProbDists").Cells(1, 4) = numS
   For i = 0 To 1000
      freqtableN(i) = 0
      freqtableQ(i) = 0
      Worksheets("ProbDists").Cells(i + 2, 1) = i
      Worksheets("ProbDists").Cells(i + 2, 3) = 0
      Worksheets("ProbDists").Cells(i + 2, 5) = i
      Worksheets("ProbDists").Cells(i + 2, 7) = 0
   Next i
   IncN = ((c1 * N * Q) - (a1 * N))
   DecN = ((-1 * c2 * N * Q) + (a2 * N))
   IncQ = ((d1 * Q) - (g1 * Q * Q) - (b1 * N * Q))
   DecQ = ((d2 * Q) + (g2 * Q * Q) + (b2 * N * Q))
   For i = 1 To numS
      Q = 61
      N = 60
      For j = 1 To num
         diceN = Rnd
         diceQ = Rnd
         If diceN < (IncN * timediff) Then
            N = N + 1
            IncN = ((c1 * N * Q) - (a1 * N))
            DecN = ((-1 * c2 * N * Q) + (a2 * N))
            IncQ = ((d1 * Q) - (g1 * Q * Q) - (b1 * N * Q))
            DecQ = ((d2 * Q) + (g2 * Q * Q) + (b2 * N * Q))
         ElseIf diceN < ((IncN * timediff) + (DecN * timediff)) Then
            N = N - 1
            IncN = ((c1 * N * Q) - (a1 * N))
            DecN = ((-1 * c2 * N * Q) + (a2 * N))
            IncQ = ((d1 * Q) - (g1 * Q * Q) - (b1 * N * Q))
            DecQ = ((d2 * Q) + (g2 * Q * Q) + (b2 * N * Q))
         End If
```

```
            If diceQ < (IncQ * timediff) Then
                Q = Q + 1
                IncN = ((c1 * N * Q) - (a1 * N))
                DecN = ((-1 * c2 * N * Q) + (a2 * N))
                IncQ = ((d1 * Q) - (g1 * Q * Q) - (b1 * N * Q))
                DecQ = ((d2 * Q) + (g2 * Q * Q) + (b2 * N * Q))
            ElseIf ((diceQ < ((IncQ * timediff) + (DecQ * timediff))) And (Q > 1)) Then
                Q = Q - 1
                IncN = ((c1 * N * Q) - (a1 * N))
                DecN = ((-1 * c2 * N * Q) + (a2 * N))
                IncQ = ((d1 * Q) - (g1 * Q * Q) - (b1 * N * Q))
                DecQ = ((d2 * Q) + (g2 * Q * Q) + (b2 * N * Q))
            ElseIf (diceQ < ((IncQ * timediff) + (DecQ * timediff) + (lam * timediff))) Then
                Q = Q - disas
                If Q < 1 Then
                    Q = 1
                End If
                IncN = ((c1 * N * Q) - (a1 * N))
                DecN = ((-1 * c2 * N * Q) + (a2 * N))
                IncQ = ((d1 * Q) - (g1 * Q * Q) - (b1 * N * Q))
                DecQ = ((d2 * Q) + (g2 * Q * Q) + (b2 * N * Q))
                numD = numD + 1
            End If
        Next j
        freqtableN(N) = freqtableN(N) + 1
        freqtableQ(Q) = freqtableQ(Q) + 1
        Worksheets("ProbDists").Cells(N + 2, 3) = freqtableN(N)
        Worksheets("ProbDists").Cells(Q + 2, 7) = freqtableQ(Q)
    Next i
End Sub
```