

## Bug Dynamics: An Example of Chaotic Population Growth and Collapse

**Learning goal:** Students will experience some key properties of nonlinear systems, discrete mappings and displaying data, in particular, how to identify bifurcations and chaos.

### Learning objective

Science content objective:

- Students will understand how the time dependence of a population is affected by the population's interactions with its environment, in particular the interdependence of living organisms and their interactions with the components of a biosphere.

Science model/computation objectives:

- Students will learn how to develop a simple model of a system that can then be computed and used to create simulated bug population data. Characteristics in the simulated data can then be compared to observed phenomena.
- Students will develop computational scientific thinking skills by developing a model with successively higher levels of complexity, and by using visualization to uncover simplicity within the complex behaviors found in the model's output.
- Students will learn how mathematical models can be used to describe the relationships among data points.
- Students will understand that computer models can be used to explore and develop scientific concepts.

Scientific skills objective:

- Students will practice the following scientific skills:
  - Graphing (visualizing) data
  - Describing trends revealed by data
  - Proportional and nonproportional (nonlinear) reasoning
  - Understanding discrete, in contrast to continuous, processes
  - Test (confirm, modify or reject) hypothesis using analyzed data

### Activities

In this lesson, students will:

- Extract data from a table.
- Identify key objects/components within a model and study their relationships.
- Make statements that describe characteristics of the system.
- Make concept maps that incorporate components and relationships.
- Create mathematical equations that describe the relationships that exist within the system statements.
- Understand the use of recursion to propagate a solution at the present time forward to that at future times.
- Create or modify computer simulations of the model and analyze their predictions compared to observed natural phenomena.

- Recognize some characteristics of nonlinear systems, such as bifurcations, intermittency and chaos.
- Be able to construct a plot of attractors versus growth rate (a “bifurcation plot”).

### **Products**

At the end of this module students will produce and evaluate 1) a list of system statements presented, 2) concept maps, 3) mathematical equations and 4) a mathematical model implemented in one of three software applications. Each product is a different view of the data and phenomenon.

### **Where’s Computational Scientific Thinking**

Students will learn how to develop abstract representations of growth properties of a population rather than focusing on individual members.

In this lesson, students will devise and explore a model for which there is no obvious algebraic solution, yet be able to determine its predictions via computation of a recursion relation.

Students will learn that the computer can be used as an experimental lab to see what happens in nature, i.e., when the growth rate is increased.

Students will discover that mathematical models can provide understanding of various types of periodic and chaotic behaviors that are not at all obvious from the start.

### **Intuition Development**

Students should be able to develop an understanding that the model will produce oscillations, and thus build intuition and insight. Understanding that simple mathematics can underlie some very complicated behavior is insightful. Bifurcation diagrams of a similar sort as developed in the module occur for many nonlinear systems.

Chaos itself occurs in many natural systems, with turbulence in water being a common example. In all cases of chaos, the mathematical description of the system must be nonlinear (a quadratic relation between present and future populations) and there must be three or more variable or parameters (Poincaré–Bendixson theorem).

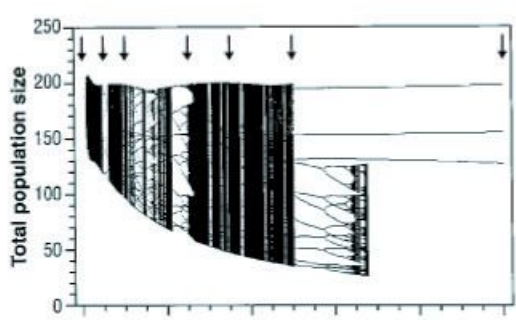
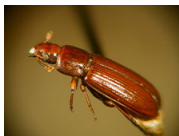


Figure 1 The population as a function of a growth rate parameter observed in the experiment of [Bugs]. Graphs such as this are called bifurcation diagrams in reference to the splitting (pitchforks) of a population into two.

## 1. Observations

We previously explored population growth and decay without limits. These are simple mathematically, but lack some of the limits that the real world often places on systems. For example, many animal species appear to have populations that remain essentially constant from year to year. For instance, the number of frogs in a pond might be 30 one year and 32 another, which is essentially constant. Clearly, a constant population number means that the number of births in a year (the birth rate) equals the number of deaths in a year (death rate). While births obviously arise from mating, deaths may arise from various causes such as aging, disease, predation, environmental stress and starvation. If there were no deaths, and if the number of births in a year were *proportional* to the number of animals present, then the population would



increase exponentially with no limit (we describe exponential behavior in another module). This actually happened, to the eventual detriment to the Australian continent, when a marine toad was introduced into Australia to eat beetle grubs.

Usually, the increase in deaths as a population gets larger limits the growth of a population, and often a maximum level is reached that proves to be rather stable. How can we represent the growth or change in the population in order to predict its change in the future?

## 2. Problem

The flour beetle *Tribolium* pictured here has been studied in a laboratory in which the biologists experimentally adjusted the adult mortality rate (number dying per unit time). For some values of the mortality rate, an equilibrium population resulted. In other words, the total number of beetles did not change even though beetles were continually being born and dying. Yet, when the mortality rate was increased beyond some value, the population was found to undergo periodic oscillations in time. Under some conditions, the variation in population level became *chaotic*, that is, with no discernible regularity or repeating pattern. **Why do we get such**

## **different-looking patterns of population dynamics? What general mathematical model could produce both sets of patterns?**

In the words of the scientists themselves:

“The mathematical theory of nonlinear dynamics has led population biology into a new phase of experimental and theoretical research. Explanations of observed fluctuations of population numbers now include dynamical regimes with a variety of asymptotic behaviors: stable equilibria, in which population numbers remain constant; periodic cycles, in which population numbers oscillate among a finite number of values; quasiperiodic cycles, which are characterized by aperiodic fluctuations that are constrained to a stable attractor called an invariant loop; and chaos, where population numbers change erratically and the pattern of variation is sensitive to small differences in initial conditions.”

A population pattern that changes radically in time due to small shifts in environmental conditions is considered chaotic. Explaining such behavior is an example of a problem that the traditional algebraic and analytic math taught in high school cannot model, yet for which computers are well suited.

**Your problem** is to develop and test the simplest mathematical model you can think of for a population that contains growth, yet also contains a mechanism to limit growth so that it remains below a maximum value. Once you have this model, you should then explore whether it can produce the types of behaviors described in the paragraph above.

## **3. From Biology to Math**

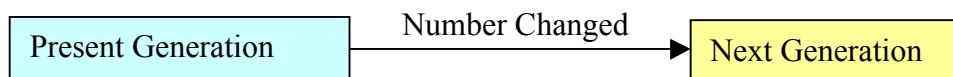
### **3.1 System Statements, Key Components and Their Relationships**

System statements are brief sentences that describe a phenomenon. They define how components of the system relate to each other. We start by determining the **components**. **For the bugs example, the components of the system are bugs and their interactions.** Next we link the components by hypothesizing what might be the relationships (processes) among the components that make them change. This is the process of writing **system statements**. Each person can have slightly different system statements as they *verbally* describe a phenomenon. The main purpose of the system statements is to bring us closer to the mathematical description of a phenomenon. The simpler the system statements are, the easier it will be to convert them into mathematics. An example of system statements for the phenomenon for the bugs system would be:

1. We start with a certain number of bugs that are currently alive, the present generation.
2. In the next generation, the change in the number of bugs due to births and deaths depends on the number of bugs in the previous generation.
3. Births clearly arise from breeding. Deaths arise from factors such as aging, disease and predation. Death rates tend to increase as populations grow, particularly due to increased competition for food and increased rates for disease. Because the change in population arises from both births and deaths, the change will be positive when births outnumber deaths, and negative when deaths outnumber births.

4. We lump births and deaths together into the simple statement that “the population tends not to grow beyond some maximum value”, without any further explanation.
5. We calculate the change in population, add it to the present population, and thus obtain the population for the next generation.
6. We call the *next* generation’s population the *present* generation’s population, and start the process over again.
7. Once we have a mathematical expression that relates the next generations’ population to the present population, we use the computer to predict the population for all latter times.

We can express these statements graphically via the concept map below. What is not visible in the map is how we represent time. We substitute the term “generation” for “population” to include reference to the passing of a time period in which the population changes.



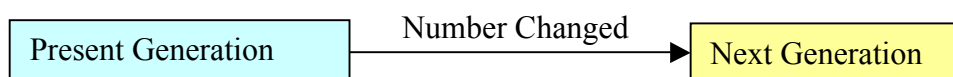
### 3.2 Underlying Theory

We postulate or theorize 1) that a balance between the rates of bug breeding and dying leads to a stable population, and that 2) there is a maximum population number for each set of rates. Specifically, we postulate that the overall rate of change in population decreases as the maximum population is reached.

Note: To keep our model simple, we will not concern ourselves with the actual causes and dynamics of dying, but rather just impose the environmental constraint that there is a maximum number of bugs possible. The effects observed in this simple model should also be present in more complete models.

### 3.3 Building a Concept Map of Model as Prelude to Equations

Concept maps graphically describe the system statements. We are going to break down the process of creating a model, writing mathematical equations, into two steps. First, let us consider the statement “The population changes from one generation to the next”.



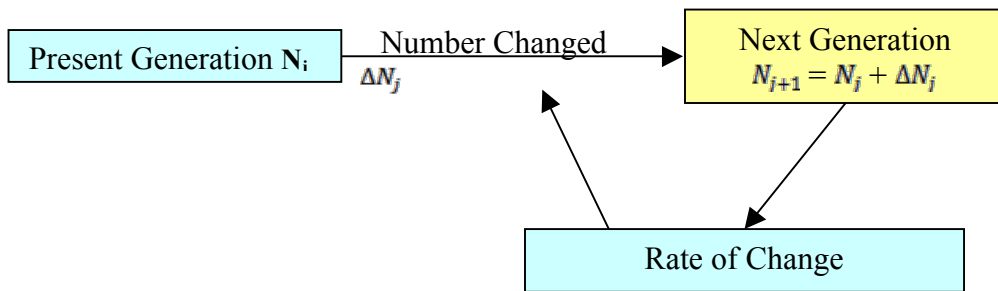
On the left we have a box that represents the number of bugs living now, during the present generation. The generation time differs for different species, with a year being typical for bugs. (For humans it might be 20 years, with the “population” of that generation being the average over the entire generation time). The average number of bugs in the next generation is obtained

by adding in the total change, which includes both births and deaths. And since some generations may have more deaths than births, the change can be a negative number.

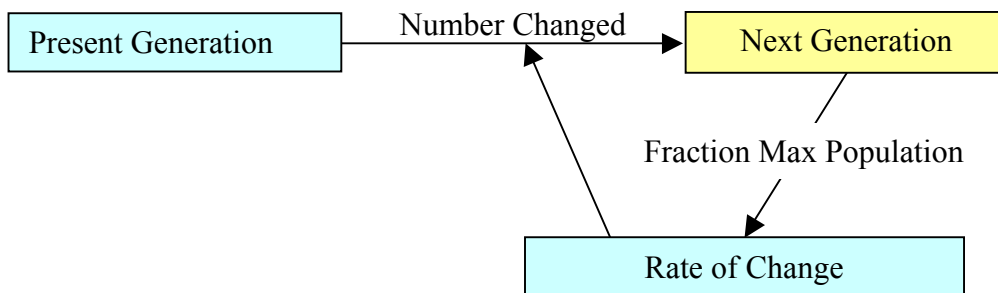
In terms of an equation, let us say the number of bugs we start with is  $N_0$ , that in the next generation it is  $N_1$ , that in the second generation it is  $N_2$ , and so forth. The diagram above states that *present + change = next*. If stated in reverse order, the concept map is *next = present + change*. In terms of symbols, we express the diagram above as the equation

$$(1) \quad N_{j+1} = N_j + \Delta N_j$$

Here  $N_j$  is the number in the present generation ( $j$ -th),  $N_{j+1}$  is the number in the next generation ( $j+1$ ), and  $\Delta N_j$  is the total change that takes generation  $j$  to generation  $j+1$ .



The heart of the model, of course, is what we use for the change  $\Delta N_j$ . We know from our module *Spontaneous and Exponential Decay*, that if the increase in population were proportional only to the number present, then we would get exponential growth or decay. We want to somehow build in the fact that the population has a maximum value. We imagine that the *Total Change* should decrease in magnitude as we get closer to reaching the maximum population number. We might show this graphically, as step two, by indicating that the *Number Changed* depend upon the rate of change (the change per generation), and that the rate changes as we get closer to the maximum population  $N_{\max}$ :



### 3.4 From a Concept Map to a Mathematical Model (With Derivation)

It is often too hard to try to guess the final form of a model all in one fell swoop. So let's start with what we already know, namely our work in the *Spontaneous and Exponential Decay*

module, and try to extend it to bug populations. The extension is based on the relations in the concept map above, and essentially adds mathematical symbols to the elements in the figures.

As we did in the *Spontaneous and Exponential Decay* module, we measure time in discrete intervals of  $\Delta t$ , and label each step or generations by a discrete integer  $j = 0, 1, 2, \dots$  :

$$(2) \quad t = j \Delta t$$

Accordingly, the number of bugs present at time  $t = j \Delta t$  is called  $N_j$  :

$$(3) \quad N_j = N(t = j \Delta t)$$

Recall, if the change in the number of atoms during a time interval were a constant  $\lambda$ ,

$$(4) \quad \Delta N_j = \lambda$$

then that number would increase *linearly* with time

$$(5) \quad N(t) = \lambda t$$

Here the proportionality constant  $\lambda$  is called the *growth rate* and has the dimension of *number per unit time*, which is the same as just *per unit time*. If, instead, the change in number  $\Delta N$ , in addition to being proportional to the length of the time interval, were also proportional to the number of species present during the time interval,

$$(6) \quad \Delta N_j = \lambda N_j \Delta t,$$

then, exponential growth results,

$$(7) \quad N(t) = N(0)e^{\lambda t}$$

where  $N(0)$  is the number present at time  $t=0$ . The important point here is that exponential growth results from the change or rate being proportional to the number present.

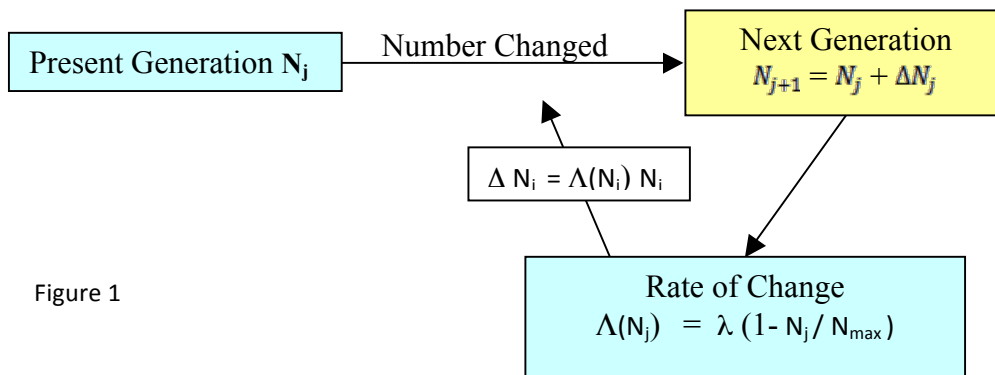


Figure 1

Equation (6) is a good starting model for our model of bug populations. With the change proportional to the number present, we would obtain exponential growth in the number of bugs with growth rate  $\lambda$ . Yet we want the model to contain a mechanism such that growth rate slows down as the number of bugs reaches its maximum  $N_{max}$ , the *carrying capacity* of the

environment. And so, as shown in the concept map above, we extend the model by postulating that the growth rate parameter, which we now call  $\Lambda(N_j)$ , is no longer a constant but instead decreases as the number of bugs  $N_j$  gets close to  $N_{\max}$ :

$$(8) \quad \Lambda(N_j) = \lambda(1 - N_j / N_{\max})$$

Here  $\lambda$  is still a constant, which can be thought of as a growth rate, while  $\Lambda(N_j)$  is now a function of the variable  $N_j$ , which incorporates births and a population limit. Indeed, if  $N_j$  become greater than  $N_{\max}$ , then Eq. (8) leads to a negative value for  $\Lambda(N_j)$ , which would mean a decrease in the number of bugs (death). Our expression then for the change in the number of bugs in generation  $j$  is

$$(9) \quad \Delta N_j = \Lambda(N_j) N_j \Delta t$$

$$(10) \quad = \lambda(1 - N_j / N_{\max}) N_j \Delta t.$$

The full model, which is also the algorithm we use for the calculation, is then

$$(11) \quad N_{j+1} = N_j + \lambda(1 - N_j / N_{\max}) N_j$$

where we have set  $\Delta t = 1$ , which means that we measure time in units of *generations*.

### 3.5 From a Concept Map to a Mathematical Model (Without Derivation)

In Figure 1 we present a concept map with mathematical representations of each step included. Our goal is to turn this concept map into a mathematical model that can be simulated on a computer using a spreadsheet such as Excel, a programming language such as Python, or a specialized modeling program like Venism. In each case we need to convert the concept map into mathematical statements that we can then use as an algorithm to program a computer.

Let's try to understand every step of this process.

$N_j$ : is the "number of bugs" in our current generation.

$\Delta N_j$ : is the "change in number of bugs" as we go from the present generation into the next generation. This is just the number of bugs born minus the number of bugs who have died.

$\lambda$ : is the growth or birth rate parameter, similar to that in our exponential growth and decay model. However, as we extend our model it will no longer be the actual growth rate, but rather just a constant that tends to control the actual growth rate without being directly proportional to it.

$\Lambda(N_j) = \lambda(1 - N_j / N_{\max})$ : is our model for the effective "growth rate", a rate that decreases as the number of bugs approaches the maximum allowed by external factors such as food supply,



disease or predation. (You can think of  $\lambda$  as the growth or birth rate in the absence of population pressure from other bugs.) We write this rate as  $\Lambda(N_j)$ , which is a mathematical way of saying  $\Lambda$  is affected by the number of bugs, i.e., “ $\Lambda$  is a function of  $N_j$ ”. It combines both growth and all the various environmental constraints on growth into a single function. This is a good approach to modeling; start with something that works (exponential growth) and then modify it incrementally, while still incorporating the working model.

$N_{j+1} = N_j + \Delta N_j$  : This is a mathematical way to say, “The new number of bugs equals the old number of bugs plus the change in number of bugs”.

$N_j/N_{\max}$ : is what fraction a population has reached of the maximum carrying capacity allowed by the external environment. We use this fraction to change the overall growth rate of the population. In the real world, as well as in our model, it is possible for a population to be greater than the maximum population (which is usually an average of many years), at least for a short period of time. This means that we can expect fluctuations in which  $N_j/N_{\max}$  is greater than 1.

By taking a close look at the concept map, we see that it can be expressed as the mathematical relation giving the change in the number of bugs in the next ( $N_{j+1}$ ) and the present generations:

$$(12) \quad N_{j+1} = N_j + \lambda (1 - N_j / N_{\max}) N_j$$

$$(13) \quad = N_j (1 + \lambda) - \lambda N_j^2 / N_{\max}$$

This is an expression that the number of bugs in a new generation is the sum of the number of bugs in the previous generation and change in bug population between the two generations.

### 3.6 Discussion of Model

Eq. (11) is fairly simple, but because there is an  $N_j^2$  term on the right-hand-side of the equation it is a *nonlinear* equation. As we shall see, that does lead to some interesting behaviors. However, because  $N_{\max}$  is expected to be a very large number, if the bug population  $N_j$  remains small, then the nonlinear term is expected to be small and we should obtain something close to exponential growth. In fact,  $\Lambda = 0$  when  $N_j = N_{\max}$ , at which point the population does not change. At the other extreme, when  $N_j$  is larger than  $N_{\max}$ , then  $\Lambda$  becomes negative, and we should get *decay* back towards  $N_{\max}$ . So we would expect that the bug population would settle at a stable population, at which point there would be neither growth nor decay. That population should be *stable* since a small variation from  $N_{\max}$  leads to either growth or decay back to  $N_{\max}$ .

Just by looking at Eq. (11) there is no way to know how good a model it is. Being as simple as it is, the model is not expected to be a complete description of the population dynamics of bugs. However, if it exhibits some features similar to those found in nature, then it may well form the foundation for a more complete description. Specifically, the model has no details about the biology of bugs or of the environment, other than the carrying capacity and the birth rate. Nevertheless, now that we have a simple model in hand as Eq. (11), as scientists, we will go ahead and explore its solutions and then compare them to nature.

Equation (11) is a form of what is known as the *logistic map* or equation. It is a *map* because it "maps" the population in one year into the population of the next year. It is "logistic" in the military sense of supplying a population with its needs. It is a nonlinear equation because it contains a term proportional to  $N_j^2$  and not just  $N_j$ . The logistic map Eq. (11) is also an example of discrete mathematics. It is discrete because the time variable  $j$  assumes just integer values, and consequently the variables  $N_{j+1}$  and  $N_j$  do not change continuously into each other, as would a function  $N(t)$ . In addition to the *variables*  $N_j$  and  $j$ , Eq. (11) also contains the two *parameters*  $\lambda$ , the growth rate, and  $N_{max}$ , the maximum population. You can think of these as "constants" whose values are determined from external sources and remain fixed as one year of bugs gets mapped into the next year. However, as part of viewing the computer as a laboratory in which to experiment, and as part of the scientific process, you should vary the parameters in order to explore how the model reacts to changes in them.

### 4.0 Real Data

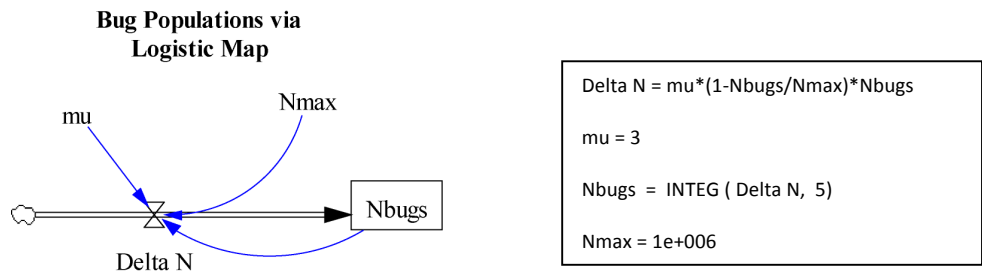
In order to validate a model, it is useful to compare its prediction to some experimental data. The following are empirical data on mortality rates and population fluctuation [A&M]:

Table 1. Cyclic variations in the abundance of forest insect species

Host insect species	Locality	Period of cycles in population abundance (years)	Pathogen	Natural mortality rate (per week)	Disease induced mortality rate (per week)
Malacosoma disstria (tent caterpillar)	North America	8 – 12	Nuclear polyhedrosis virus	0.07	0.37

### 5. Implementations

5.a Vensim: BugDynamicsSampleVensim.mdl



We see from the Vensim equations in the box that the simulation integrates or sums (INTEG) the bug population, with a change of Delta N in each generation, starting with an initial value of 5. The equation for DeltaN is a version of Eq. (11), the maximum population is set to be one million, and the growth rate constant  $\mu = 3$ .

5.b Python: Bugs.py

# Bugs.py creates bifurcation diagram for Logistic Map from visual.graph import \*

m\_min = 1.0; m\_max = 4.0; step = 0.025

lasty = int(1000 \* 0.5) # to eliminate later some points  
 count = 0 # to plot later every two iterations  
 pts = gdots(shape = 'square', size = 1, color = color.green)

```
for m in arange(m_min, m_max, step):
    y = 0.5
    for i in range(1,401,1): # to avoid transients
        y = m*y*(1-y)
    for i in range(201, 402, 1):
        y = m*y*(1 - y)
        inty = int(1000 * y)
        pts.plot(pos=(m,y))
    lasty = inty
    count += 1
```

graph1 = gdisplay(width = 600, height = 400, title = 'Logistic map', xtitle = 'm', ytitle = 'x', xmax = 4.0, xmin = 1.0, ymax = 1.0, ymin = 0.0)

## 5.1 Excel: BugDynamicsSampleSpreadsheet.xls

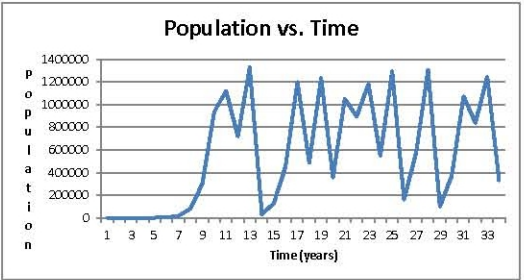
Year	Change In Population From Previous Year	New Population
0		5
1	14.999925	19.999925
2	59.99857501	79.99850001
3	239.9763007	319.9748008
4	959.6172506	1279.592051
5	3833.864087	5113.456138
6	15261.92611	20375.38225
7	59880.67815	80256.0604
8	221445.0755	301701.1359
9	632032.6815	933733.8174
10	185624.9269	1119358.744
11	-400815.7625	718542.9818
12	606716.8953	1325259.877
13	-1293161.594	32098.28283
14	93203.94921	125302.232
15	328804.7481	454106.9801
16	743681.4922	1197788.472
17	-710726.2562	487062.2161
18	749497.8412	1236560.057
19	-877562.1542	358997.9031
20	690355.226	1049353.129
21	-155366.5817	893986.5475
22	284323.8011	1178310.349
23	-630314.7872	547995.5614
24	743089.2782	1291084.84
25	-1127445.671	163639.169
26	410584.174	574223.343
27	733472.6861	1307696.029
28	-1207118.626	100577.409
29	271384.7669	371962.1699
30	700818.9422	1072781.112
31	-234234.6069	838546.5051
32	406158.7916	1244705.297
33	-913757.937	330947.3597

Growth-Death Rate Variable	Carrying Capacity of Environment (Nmax)
	1000000

Notes on how to use:  
 Try changing the "Growth-Death Rate Variable" in cell F2.  
 Using numbers (including decimals) between 1 and 4 should give you radically different population curves.

To try:

- Find a "Growth-Death Rate Variable" that results in a stable population curve.
- Find a "Growth-Death Rate Variable" that results in a chaotic population curve.
- Find a "Growth-Death Rate Variable" that results in a physically unrealistic population curve.
- For each of the above, write a paragraph that describes a possible physical scenario for the population of a bug due to its growth-death rate and environmental constraints.



Explaining the Equation:

$$\Delta N = \lambda \left( 1 - \frac{N}{N_{\max}} \right) N$$

Notes on spreadsheet usage:

1. Try changing the "Growth-Death Rate Variable" in cell F2.
2. Using different values (including decimals) for this rate between 1 and 4 should give radically different populations curves.
3. Try to find a rate that results in a static population curve (constant in time).

4. Try to find a rate that results in a chaotic population curve (no discernible pattern).
5. Try to find a growth rate that results in a physically unrealistic population curve.
6. For each of the above, write a paragraph that describes a possible physical scenario that might lead to this type of population curve.

## 6 Analyzing and Using Model

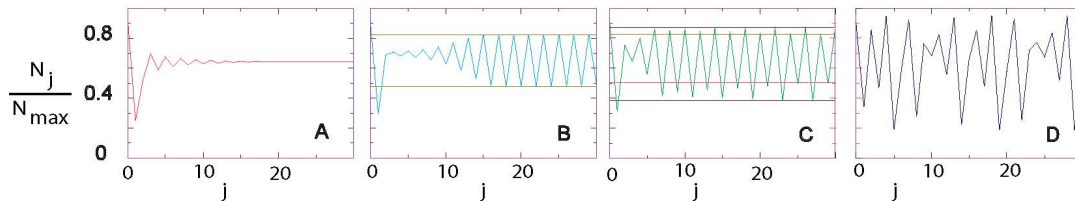


Figure 2: The insect population  $N_j$  versus the year number  $j$  for four different growth rates: (A)  $\lambda = 1.8$ , a single, stable population; (B)  $\lambda = 2.3$ , a two cycle; (C)  $\lambda = 2.5$ , a four cycle; (D)  $\lambda = 2.8$ , a chaotic regime.

### 6.1 Looking for Stable Populations (See Figure 2)

As the first exploration with our model, we want to see if it can produce a *stable* population. By “stable” we mean a population that may pass through a transient stage but eventually reaches a value at which it remains essentially the same from generation to generation. By “transient” we mean something that lasts only a short time and so is not a permanent feature.

1. We will be calculating the following equation using either a spreadsheet or a program in python. These let us compute the next-year population  $N_{j+1}$  as a function of year number  $j$  based on Equation (10).
2. To set the scale for the calculations, we will assume a value for the maximum population, say  $N_{\max} = 1000$ .
3. Since the model generates a series of populations for a given initial population, we need to specify a value for the initial population  $N_0$ . This is also called the seed population. (While this may seem like another parameter, it really is just the initial value for one of the variables in the model.) Start with a small value, say  $N_0 = 100$ , but feel free to experiment with other values.
4. The value assumed for the growth rate  $\lambda$  is very important. Too large a growth rate may lead to instabilities that “blow up”, and too small a value may lead to nothing of much interest. So let’s be careful. To make sure that the model is behaving reasonably before going exploring, let try some cases for which we can be fairly sure of what the model should predict. One clever way to do this is to try out some negative and zero values for  $\lambda$ . This should lead to a decaying population, or a static population. (A *static* population is one that does not change in time. A *stable* population is one that may experience deviations from some average value, but which always returns to this average value.) We suggest  $\lambda = -1, -0.75, -0.5, -0.25, 0$ .

5. Make plots of  $N_j$  versus  $j$  for each of these values of  $\lambda$ . Seeing pure decay, growth or no change serves in part as a validation of the model, that is, as a check that the model works the way it should in situations where we now the correct behavior.
6. Now that you have some confidence in the model, let's see if we can increase the population to some stable values. With the same initial population as before, try  $\lambda = 0, 0.5, 1.0, 1.5, 2$ . Make plots of  $N_j$  versus  $j$  for each of these values of  $\lambda$ .
7. Take note of the *transient* behavior in these plots. These transients are the various values that the population passes through before a steady state or more regular behavior sets in. (They are called transients because they the population just passes through them without settling there.)

## 6.2 Checking Validity of Model: Model Assessment

You should have found by now that with positive growth rates  $\lambda$ , this model does yield a variety of stable populations, with the bugs approaching the maximum population more rapidly for larger values of  $\lambda$ . This also serves in part as a validation of the model, that is, a test that this simple model predicts stable populations. Next, check the validity of this model, that is, that it is general and is not highly dependent upon the specific values of the parameters:

1. For fixed values of the growth rate  $\lambda$  and initial population  $N_0$ , try different values for the maximum population  $N_{\max}$ . Verify that increasing values of  $N_{\max}$  lead to proportionally larger maximum populations, but do not affect that rate at which the maximum is attained. The transients may be affected.
2. For a fixed value of maximum population  $N_{\max}$  and growth rate  $\lambda$ , try different values for the seed population  $N_0$ . Verify that differing values of  $N_0$  do not affect the values of the maximum populations, although it may take longer to reach that maximum. Again, transients may be affected, but not the values for the stable populations.
3. For several combinations of  $N_{\max}$ ,  $\lambda$  and  $N_0$ , record the number of cycles that the system passes through before a stable population is obtained. (Variations of  $\sim 3\%$  can be considered stable for this investigation.) This number will be useful when later when we create a bifurcation plot.

## 6.3 Modeling Empirical Data

In Table 1 we have presented some empirical data on the tent caterpillar. See if you can set the parameters of our model to make it consistent with those data.

## 6.4 Activities II: Exploration; Cause Some Chaos

We have seen that our model described by Equation (11) provides an explanation of how a bug population grows or decays until a stable population is reached. It is not anything fancy, but it does demonstrate how exponential growth can be moderated. Well, rather than declare "mission accomplished" and go out for a beer, let's try to experiment with our model and see what happens as we increase the value of the growth rate  $\lambda$ . This is doing experimental science with the computer as our laboratory.

1. Start with values of your own choosing for the initial and maximum populations  $N_0$  and  $N_{\max}$ . You can verify latter that the interesting dynamics that occur are independent of these choices.
2. Produce a graph of population  $N_j$  versus year number  $j$  for  $\lambda = 2.3$ . This should produce what is called a two cycle, in which, after some transient behavior, the population oscillates between two values. Populations with a single cycle are called stable because they return to that population value when slightly disturbed. A two cycle population, in contrast, cannot be called stable because one of the populations changes to a significantly different one in the next year. However, because the system acts as if it were attracted to these two populations, these populations are called attractors or cycle points.
3. See if you can increase the value of  $\lambda_b$  from a value of 1, which produces a single cycle, upward until a two-cycle first occurs. One says that the system's long-term population bifurcates into two populations at this point, an effect known as period doubling or bifurcation.
4. Produce a graph of population  $N_j$  versus year number  $j$  for  $\lambda = 2.5$ . This should produce what is called a four cycle population, which, after transients, oscillates between four values.
5. Produce a graph of population  $N_j$  versus year number  $j$  for  $\lambda = 2.8$ . This should produce a population that oscillates between many values, with no discernable pattern. Such behavior is called chaotic. In an operational sense, a chaotic system is one with an extremely high sensitivity to the precise values of the parameters or initial conditions. This sensitivity to even minuscule changes is so high that it is impossible to predict the long range behavior unless the parameters are known to infinite precision (a physical impossibility).

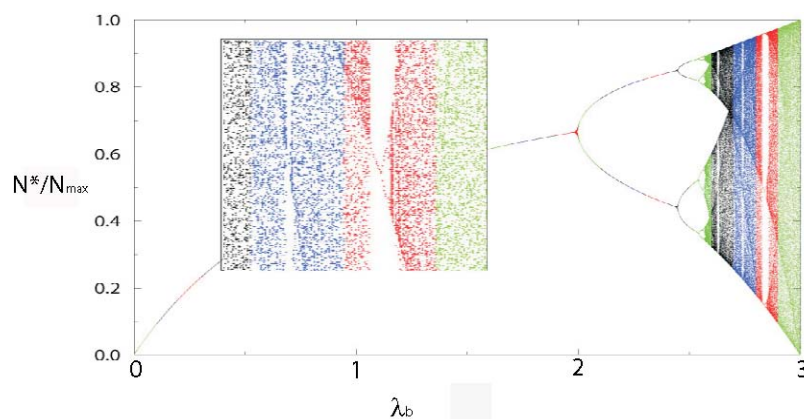


Figure 3: A bifurcation plot of attractor populations versus growth rate  $\lambda_b$  for the logistic map. The attractor populations have been divided by the maximum population in order to make the figure more universal.

1. Observe simulations for  $2.8264 < \lambda < 2.8304$ . In this range the system should exhibit what is called intermittency. Specifically, it should appear stable for a finite number of years and then jump all around, only to become quasi stable again.
2. Chaotic systems are known to display an extremely high sensitivity to the exact values of the parameters used in the calculation. Repeat the calculation of  $N_j$  versus  $j$  for a

$\lambda$  value that led to chaotic behavior. Do this for two nearly identical values of  $N_0$ :  $N_0 = 750$  and  $N_0 = 750.00000001$ . Plot the two results on the same graph. You should see the two results start off essentially identically, but eventually separate. This is an example of the “butterfly effect” in chaotic systems; presumably a butterfly flapping its wings somewhere in South America may affect the weather patterns in North America.

## 6.5 A Bifurcation Diagram\*

Computing and watching bug populations change with year number does give a good idea of the basic dynamics at play, at least until the oscillations get too complicated for you to discern patterns. In particular, as the number of bifurcations keeps increasing and the system approaches chaos, it is hard for us to see a simple underlying structure within the complicated behavior. One way to visualize what is going on is to concentrate on the attractors (also called long-term iterates), that is, those populations  $N^*$  that appear to attract the solutions and to which the solutions continuously return. A plot of these attractors as a function of the growth rate  $\lambda$ , such as shown in Figure 3, is called a bifurcation diagram and is an elegant way to summarize an extensive collection of computer calculations.

To create this type of diagram:

1. Break up the range  $0 \leq \lambda \leq 3$  into 50 steps and loop through them, either by running your program 50 times, or by extending the program so that this is done automatically (a loop).
2. Wait 200 years of model time (or whatever number you found) for the transients to die out. Then print out the next 50 values of  $(x, y)$  points  $(\lambda, N_j/N_{\max})$ , and plot them. Alternatively, you can plot directly from your program and skip the print out. We divide the ordinate (y value) by  $N_{\max}$  to present the results in a more universal form and to save work. Hint: save the values of  $N_j/N_{\max}$  to only two decimal places to avoid excessive plotting; you will not be able to see any fine details on your plots anyway.
3. Optional: Enlarge a section of your plot where where the density of points is high. Notice that similar bifurcations are contained within the magnified portion of the plot. This is called self similarity and is characteristic of fractal objects (of which sections of this plot are).

## 7. Going Beyond the Logistic Map

The logistic map includes the effect of a maximum population limit, but does not include separate rates for birth and death. In the *Predator Prey* module we extend the model by introducing a second species that feeds upon the bugs, and then observe how these populations appear to be related. You can attempt to make your own predator – prey model or look at the one in our module.



## 8. Where's the Math?

The math of exponential decay and growth is assumed to be familiar, and so should be covered before this project. The model employs discrete mathematics, and this should be contrasted with the use of continuous variables (there is also a version of the logistic map using continuous variables [Shif]). The notion of a nonlinear equation is introduced and this can be contrasted with a linear equation. Nonlinear equations often do not have closed-form, algebraic solutions. (For this simple mode, one can generate some very long algebraic expressions for the population in an arbitrary year, but they are not illuminating or practical for calculation.) In particular, even though a separate  $N_j$  appears on the right hand side of Equation (9), since an  $N_j^2$  also appears on the right hand side, we cannot say that  $N_{j+1}$  is proportional to  $N_j$ . The exploration sections deal with nonlinear dynamics, a lack of regularity and chaos, which are typical of nonlinear systems. A broad conclusion that can be drawn from this project is that simple dynamics or equations can lead to very complicated behavior. Conversely, complicated phenomena may have simple explanations, but it may take some research to uncover them.

## 9. Exact, Approximate, Precision, Accuracy, Uncertainties

The model, being so simple, is at best a crude approximation to what may be occurring in nature. We thus do not expect it to be a truly accurate description of nature. Our numerical solution of it is exact, in part because computers perform discrete mathematics exactly. If we were employing a continuous model, say one based on differential equations, then we would have to solve that only approximately on the computer. The precision of your answers depends on the precision your particular implementation (computer language). This should be fairly high (~16 places in Python and Vensim and variable in Excell, but up to 30), yet because this model exhibits chaos in certain regions, its predictions there may be hypersensitive to the exact values of all the parameters.

## 10. Summary and Conclusions

Although simple and containing few details, the model is able to predict how a population of bugs reaches a stable maximum value. We see exponential-like growth at first, followed by a decreasing growth rate as the maximum population is reached.

The explorations performed with this model discovered that larger growth rates lead to periodic behavior, with the population varying among several discrete values from year to year. Sometimes there might be only two populations that repeat, while for larger growth rates there is an increasing number of populations. With increasing growth rates, the population becomes chaotic, with no discernable pattern and with high sensitivity to the exact values of the parameters used in the model. Apparently, as the value of  $N_{max}$  is exceeded, the effective growth rate becomes negative and the population decays. However, this leads to another change in sign of the effective growth rate and the population starts to grow again. When the numbers are just right, we obtain the observed oscillatory behavior of population. Such oscillations are observed in nature, for example with cicada populations.



Discussion questions related to this project might involve birth rates, death rates, the effect of food supplies, death rates, ecological balance, and the limits to human populations. Further discussions may involve climate change and the large variations now being observed. The climate is also a nonlinear system and thus is subject to the types of irregular and chaotic behaviors observed with this model.

## References

[Bugs] Costantino, R. F., R. A. Desharnais\*, J. M. Cushing and B. Dennis, *Chaotic Dynamics in an Insect Population*, *Science*, **275**, 389-391 (17 January 1997), DOI: **10.1126/science.275.5298.389**.

[CP] Landau, R.H., M.J. Paez and C.C. Bordeianu, (2008), *A Survey of Computational Physics*, p 289-297, Princeton Univ. Press, Princeton.

[PR] Lesh, R., T. Post and M.B. Northern, *Proportional Reasoning*, [http://www.cehd.umn.edu/rationalnumberproject/88\\_8.html](http://www.cehd.umn.edu/rationalnumberproject/88_8.html); Lesh, R., Post, T., & Behr, M. (1988). *Proportional Reasoning*. In J. Hiebert & M. Behr (Eds.) *Number Concepts and Operations in the Middle Grades (pp. 93-118)*. Reston, VA: Lawrence Erlbaum & National Council of Teachers of Mathematics; *The Rational Number Project*, <http://www.cehd.umn.edu/rationalnumberproject/>.

[Ras] Rasband, S.N., *Chaotic Dynamics of Nonlinear Systems*, John Wiley, New York.

[Shif] Shiflet, A.B. and G.W. Shiflet, (2006), *Introduction to Computational Science*, Princeton Univ. Press, Princeton.

[A&M] Anderson, R. M. & May, R. M. (1980). Infectious diseases and population cycles of forest insects. *Science*, 210(4470), 658-661.