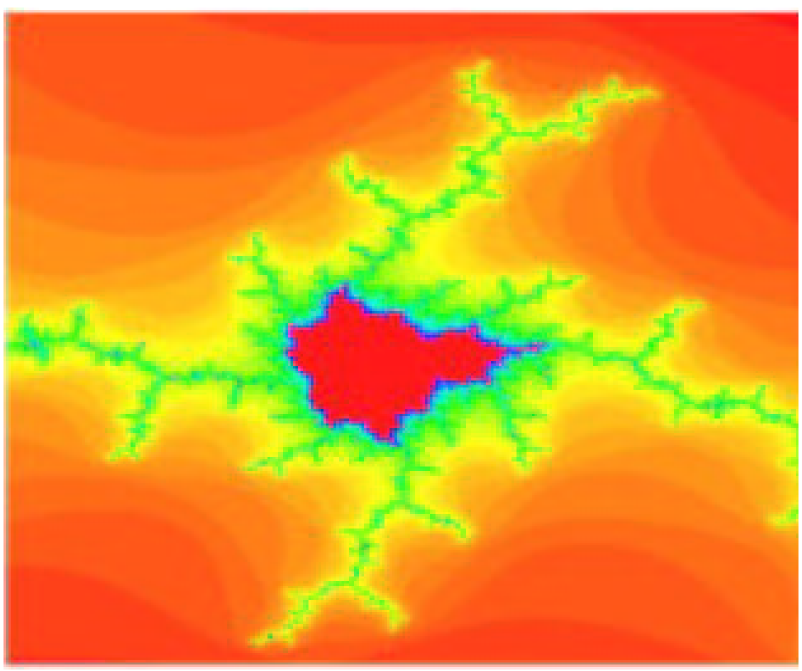
Chaotic Rendezvous:

Introductory Chaos Experimentation Using Maple Software and a Desktop Computer

by Thomas E. Oberst



Chaotic Rendezvous: Introductory Chaos Experimentation Using Maple Software and a Desktop Computer

Thomas E. Oberst

Duquesne University
Department of Physics
Pittsburgh, PA 15282

Chapter 3

The Lorenz and Hénon Attractors

In this chapter we will explore two dynamical systems: the Lorenz attractor and the Hénon attractor. They are more complicated than the quadratic, logistic, and Baker's functions of Chapters 1 and 2, but not terribly so. I believe you will find that the fascinating properties exhibited by the plots of these attractors make any additional vicissitudes you experience well worth the trouble.

Both the Lorenz attractor and the Hénon attractor are objects known as *strange attractors*. What is a strange attractor? To answer this question, I will have to introduce a couple of concepts in chaos theory which you have most likely not yet encountered, and probably won't explore fully until later in this manual. However, I believe that introducing and working with these two attractors and these new concepts will be an excellent primer for your future studies in chaos.

The definition of a strange attractor, then, is the orbit of a simple system of equations which has three main properties:

1. The orbit traced by a strange attractor is attractive. This means that the orbits of points falling close to the attractor will converge towards the orbit traced by the attractor itself. That is, the attractor "attracts" the orbits of all nearby points. (You will explore the concept of *attraction* more fully in Chapter 6.)

2. Strange attractors are extremely sensitive to initial conditions. (We know all about sensitivity to initial conditions from Chapter 2.)

3. Strange attractors are *fractals*. Fractals, which will be formally introduced in Chapter 9, are objects that show self-similarity at every level of magnification.

While a large set of initial seed values may seem to lead to orbits that trace the same strange attractor, a strange attractor is technically defined for one and only one initial seed value. All other nearby orbits may resemble and be attracted to the strange attractor, but they will never reach it. For most intents and purposes, these look-alike strange attractors which are generated for a given system are usually considered one and the same, regardless of the initial seed used. Keep in mind, then, that when you encounter pictures of strange attractors, you will most often not be looking at the true strange attractor, but a very, very similar strange attractor that is being forever attracted closer and closer to the invisible, idealized orbit which is the true strange attractor.

I think that if I say anything else about strange attractors without showing you one, I will lose you completely. So let's get started.

3.1 The Hénon Attractor

In 1962, the Frenchman Michel Hénon, then working at Princeton University, became immersed in the theoretical exploration of systems of globular clusters (astronomical clusters containing anywhere from a few hundred to a few million stars). Hénon was interested in the orbits traced by stars in these clusters acting under the gravitational influence of one another when the globular cluster is considered as a dissipative system—that is, a system in which energy is lost over time, leading to the eventual collapse of the cluster under its own gravitational powers. Hénon, working with the graduate student Carl Heiles, developed a set of equations describing the orbits of stars in a globular cluster and then created a map of these orbits as they passed through an imaginary plane placed in their path. He found that the orbits were very strange. They did not form closed paths and never repeated themselves. Some even formed figure-eights and other strange curves. He published his findings and moved on to other problems.

In 1976, working at the Nice Observatory in southern France, Hénon heard about the strange attractors of Lorenz (discussed in Chapter 2) and others and decided to take a closer look at the work he had done on globular clusters years before. He greatly simplified his equations and used newer computers to plot millions of points. The resulting plot looked like a banana. What Hénon had discovered was a new and important attractor, now known as the Hénon attractor.

The equations Hénon developed in 1976, which resulted in the creation of his attractor are:

$$\begin{aligned} x_{n+1} &= y_n + 1 - \alpha x_n^2 \\ y_{n+1} &= \beta x_n \end{aligned} \tag{3.1}$$

where α, β are constants and $\alpha, \beta \in \Re$. The second equation can also be written as $y_n = \beta x_{n-1}$. Hence, the two equations can be expressed as a single "two-generation" recursive equation:

$$x_{n+1} = -\alpha x_n^2 + \beta x_{n-1} + 1$$

Note the nonlinear αx_n^2 term, which must be present for chaos to appear. (Linear equations can also exhibit chaotic behavior, but only if a discontinuity is present, such as in the Baker's function, Equation 2.1.)

The most unusual thing about the Hénon attractor is that the equations that define it are not differential! The Hénon attractor is a *map*—just like the logistic and quadratic functions of Chapter 1. So if you successfully completed Chapter 1 (and Experiment 1.3), then you already know how to handle the Hénon. The only catch is that it is a *two-dimensional* map—something you haven't dealt with yet in this manual. Recall from Chapter 1 the methods developed for iterating one-dimensional maps in Maple. The two most important things we did were to define an array in which to store the sequence of values resulting from our iteration, and to perform the iteration using a `for` loop. It stands to reason that, since we now have *two* variables (and hence *two* initial seed values) we should use *two* arrays. One array will store our x-valued iterates. The other will store our y-valued iterates. It also stands to reason that we will need to perform *two* operations during each round of the loop. Maple will perform two or more operations during each round of a loop if all of the desired operations are included between the `do` and `od` keywords and are each separated by a semicolon (this technique was used to create "side-by-side" orbits in Chapter 2 above). Below I have created a Hénon attractor with 50,000 iterations for the parameters $\alpha = 1.4$ and $\beta = 0.3$ and the initial seed $(x_0, y_0) = (0, 0)$. Because of the large number of iterations being performed, it usually takes Maple a minute or two to execute these commands.

```
> X:=array(0..50000);
```

$$X := \text{array}(0..50000, [])$$

```
> Y:=array(0..50000);
```

$$Y := \text{array}(0..50000, [])$$

```
> alpha:=1.4;
```

$$\alpha := 1.4$$

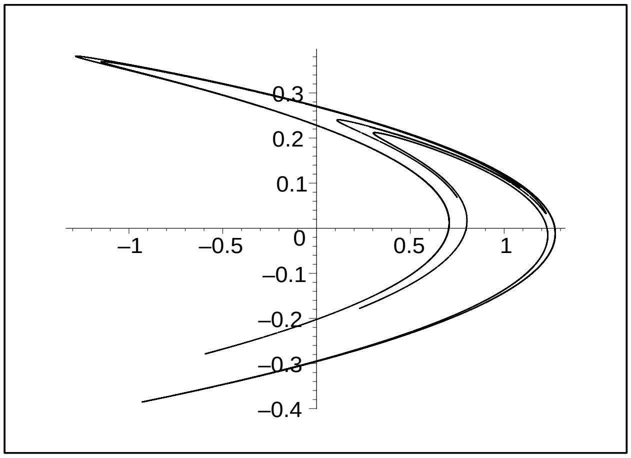
```
> beta:=.3;
```

$$\beta := .3$$

```
> X[0]:=0;Y[0]:=0;
```

$$X_0 := 0$$
$$Y_0 := 0$$

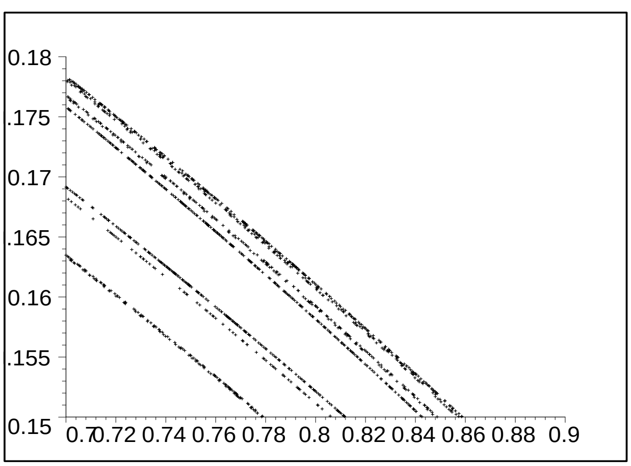
```
> for i from 0 to 49999 do X[i+1]:=Y[i]+1-alpha*X[i]^2; Y[i+1]:=beta*X[i]
od:
> pointplot({seq([X[n],Y[n]], n=20..49999)}, symbol=POINT, color=black);
```



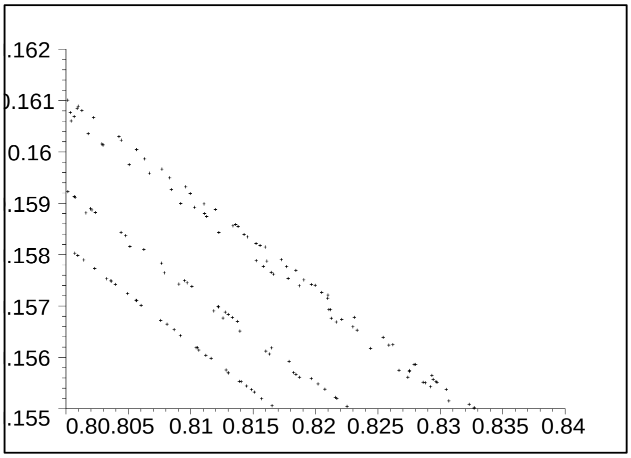
The way in which we have plotted the Hénon attractor is very different from the way in which we plotted our one-dimensional mappings in Chapter 1. Instead of plotting the iterative values versus the iteration number as our ordered-pair points, we are plotting our x and y iterate values as our ordered-pair points. Hence, the graph displayed above offers no information about which iteration a particular point belongs to. For one of the points on the graph above, all we can say for sure is that it falls on the orbit of $(x_0, y_0) = (0, 0)$ somewhere between the 20^{th} and $50{,}000^{\text{th}}$ iteration.

Since this is a strange attractor, it should demonstrate the three properties which define a strange attractor. Verifying this is part of your job in the experiments below. I would like to look at its fractal nature here, though, just for fun. (And to help you out a bit.) Below I have performed two successive magnifications of parts of the attractor. The maximum and minimum values on the axes of the plots will tell you the region I chose to magnify. Note the amazing self-similarity!

```
> pointplot({seq([X[n],Y[n]], n=20..49999)}, symbol=CROSS, color=black,
view=[(.7)..(.9),(.15)..(.18)], scaling=unconstrained);
```



```
> pointplot({seq([X[n],Y[n]], n=20..49999)}, symbol=CROSS, color=black,
view=[(.8)..(.84), (.155)..(.162)], scaling=unconstrained);
```



I have elected here to set the `symbol` option equal to `CROSS`, rather than `POINT`, which was used in the plot of the original Hénon attractor above. The reason for this is that, as I look at smaller and smaller portions of the attractor, a lesser and lesser number of points will be visible. I believe that the pattern is more lucid using the small crosses. I could have just plotted more than 50,000 points, of course. This would also have solved the problem, and perhaps allowed for yet another magnification to demonstrate another level of self-similarity. But due to the long amount of time necessary to plot the Hénon attractor, I elected not to do this.

It is also important to note that not all values of α and β will lead to strange attractors. Some values of (α, β) will lead to fixed point or periodic orbits. You can try adjusting these values and seeing what happens, but if you want a strange attractor, don't stray very far from $(\alpha, \beta) = (1.4, 0.3)$. The Hénon attractor is technically defined only as the attractor which is produced with the values $(\alpha, \beta) = (1.4, 0.3)$.[1]

3.2 The Lorenz Attractor

In 1961—around the same time Michel Hénon was working on globular clusters at Princeton—Edward Lorenz, a mathematician who had developed a savvy for examining the weather, was carrying out meteorological calculations at MIT. (Edward Lorenz is discussed in greater detail in Chapter 2.) Lorenz was using a simplified set of differential equations in time, which he had developed to model weather patterns, and which hopefully could be used to make long-term predictions. His equations are as follows:

[1]Gulick, Denny, Encounters with Chaos. New York: McGraw-Hill, Inc., 1992. Page 175.

$$\begin{aligned}\frac{dx}{dt} &= \sigma(y-x)\\ \frac{dy}{dt} &= rx-y-xz \\ \frac{dz}{dt} &= xy-bz\end{aligned} \tag{3.2}$$

Where σ, b, and r are real constants.

Lorenz's equations are certainly more complicated than Hénon's mapping! Since there are now three variables, if a strange attractor is to be birthed, its points will be ordered triplets, and hence its form can be displayed in three-dimensional space. In addition, this is our first encounter with a *flow*. The upshot of the difference between flows and maps is that the orbits of flows are *continuous*, rather than specified at fixed intervals.

Our ultimate aim is to plot the strange attractor that results from this system of equations. We could try to solve the system, specifying *exact* solutions for x, y, and z, but I doubt that we will have any luck doing this by hand, or even with Maple, for that matter. We can use Maple to generate lists of *numerical* solutions, but, because we want a plot, we would then have to figure out a way to create a three-dimensional point-plot using these lists of data. I am not saying that this approach is impossible. It can be done. However, all of this can be accomplished in a single step, using the high-powered `DEplot3d` command.

`DEplot3d` is located in the package `DEtools`, so the first thing we must do is load this package using the `with` command. The argument of `DEplot3d` accepts many members, which must be ordered as follows: A set of ordinary differential equations enclosed in curly brackets, a set of dependent variables which you wish to solve for enclosed in curly brackets, the range of the independent variable, one or more lists of initial conditions with each list of initial conditions enclosed in square brackets and then the entire list of lists of initial conditions enclosed in square brackets, and other options. Many of the other options available are similar to those specified in Table 3 in the Introduction to Maple above, such as `axes`, `scaling`, `style`, `title`, etc. Some new options are available as well, including `scene` and `stepsize`; `scene` specifies the plot to be viewed. For example, `scene` $= [x, y, z]$ indicates that the plot of x versus y versus z is to be plotted, with t implicit, while `scene` $= [t, y, z]$ plots t versus y versus z (t explicit). For our purposes, we will want to set `scene` $= [x, y, z]$. Although the exact solution to Lorenz's set of differential equations is continuous, `DEplot3d` uses specific calculated points belonging to the solution curve of the the system to generate a plot. The `stepsize` option gives us control over how many of these points to include within the specified range of the independent variable. (In this way it is similar to the `numpoints` command used for normal plotting.) The smaller you make your `stepsize`, the higher the number of points you will be including and the more the plot will appear to be continuous, but the longer you will have to wait for Maple to generate the plot. If you elect not to use this option, the default is `stepsize` $= \frac{|b-a|}{20}$, where $a...b$ is the range of the independent variable. For more information on these options, search under the Maple *Help* menu for `DEplot3d`.

Here's how it's done (using the constants $\sigma = 10$, $r = 28$, $b = \frac{8}{3}$ and the initial conditions $(x(0), y(0), z(0)) = (0.1, 1, 1)$):

```
> with(DEtools):
> sigma:=10; r:=28; b:=8/3;
```

$$\sigma := 10$$
$$r := 28$$
$$b := \frac{8}{3}$$

```
> L1:=diff(x(t),t)=sigma*(y(t)-x(t));
```

$$L1 := \frac{\partial}{\partial t}\,\mathrm{x}(t) = 10\,\mathrm{y}(t) - 10\,\mathrm{x}(t)$$

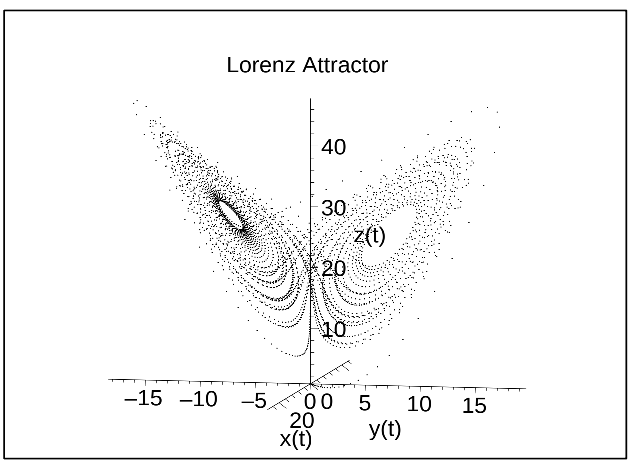
```
> L2:=diff(y(t),t)=r*x(t)-y(t)-x(t)*z(t);
```

$$L2 := \frac{\partial}{\partial t}\,\mathrm{y}(t) = 28\,\mathrm{x}(t) - \mathrm{y}(t) - \mathrm{x}(t)\,\mathrm{z}(t)$$

```
> L3:=diff(z(t),t)=x(t)*y(t)-b*z(t);
```

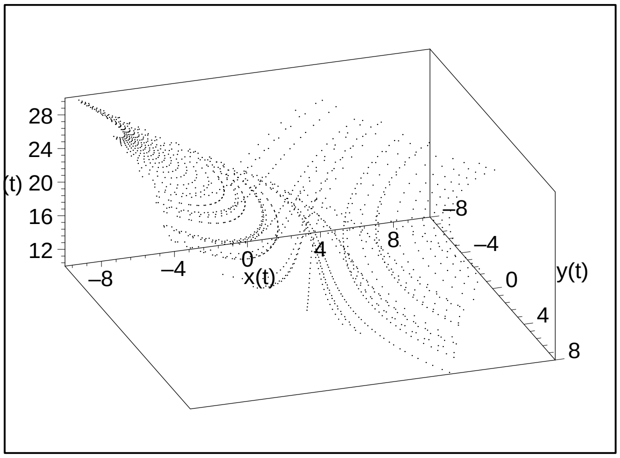
$$L3 := \frac{\partial}{\partial t}\,\mathrm{z}(t) = \mathrm{x}(t)\,\mathrm{y}(t) - \frac{8}{3}\,\mathrm{z}(t)$$

```
> DEplot3d({L1,L2,L3}, {x(t),y(t),z(t)}, t=0..50, stepsize=.015,
[[x(0)=.1,y(0)=1,z(0)=1]], scene=[x(t),y(t),z(t)], axes=normal, shading=XYZ,
title="Lorenz Attractor", style=point, symbol=POINT, scaling=unconstrained,
orientation=[-101,100]);
```



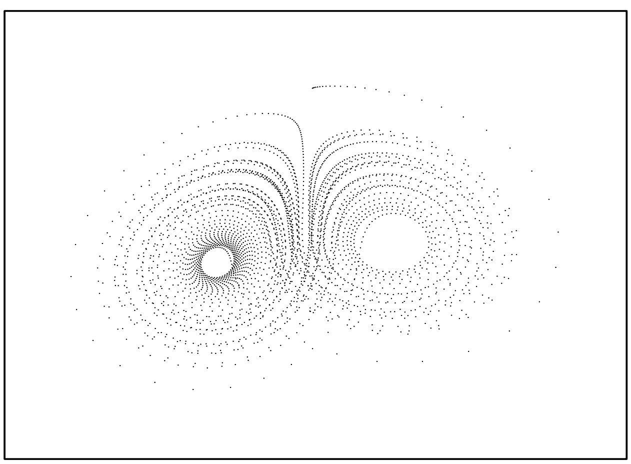
After you create this attractor, click on the plot on your Maple worksheet so that a black border appears around the plot. Now, with your mouse, click on the Lorenz attractor again and drag your mouse to spin the attractor around, upside down, or to any orientation that you wish to view it from! (This plays a role commensurate to the `orientation` option, which I have included in my Maple entries here.) Note the intricate weaving that occurs between the two "wings" of the attractor. We can adjust our `DEplot3d` options to get a closer and more revealing view of these weavings:

```
> DEplot3d({L1,L2,L3}, {x(t),y(t),z(t)}, t=0..50, stepsize=.015,
[[x(0)=.1,y(0)=1,z(0)=1]], view=[-10..10, -8..8, 10..30],
scene=[x(t),y(t),z(t)], axes=boxed, shading=XYZ, style=point, symbol=POINT,
scaling=unconstrained);
```



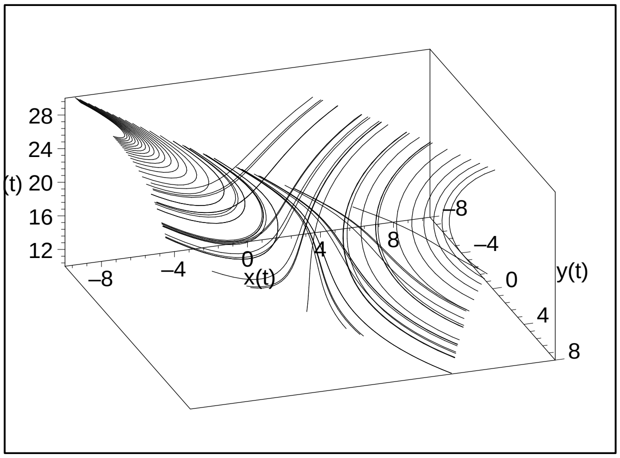
We can also plot the Lorenz attractor without axes to block the view:

```
> DEplot3d({L1,L2,L3}, {x(t),y(t),z(t)}, t=0..50, stepsize=.015,
[[x(0)=.1,y(0)=1,z(0)=1]], scene=[x(t),y(t),z(t)], axes=none, shading=XYZ,
style=point, symbol=POINT, scaling=unconstrained, orientation=[-13,-100])
;
```

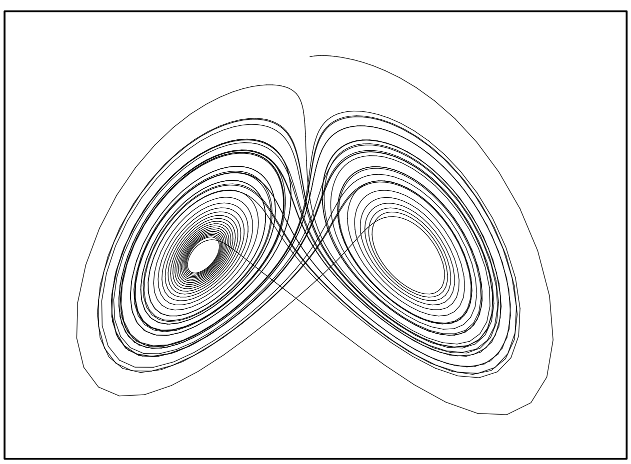


To be truer to the actual continuous nature of the Lorenz attractor, we can set our `style` option equal to "line." This will instruct Maple to connect the points of the attractor in an orderly fashion. It is still important, however, to choose a proper `stepsize`. If you don't, then your lines may look like the choppy connect-the-dot pictures in children's books rather than smooth curves. Also, Maple seems to have a will to plot the Lorenz attractor with extremely thick lines. Therefore, you should also include the `thickness` option to give yourself direct control over this parameter:

```
> DEplot3d({L1,L2,L3}, {x(t),y(t),z(t)}, t=0..50, stepsize=.015,
[[x(0)=.1,y(0)=1,z(0)=1]], view=[-10..10, -8..8, 10..30],
scene=[x(t),y(t),z(t)], axes=boxed, shading=XYZ, style=line, thickness=0,
scaling=unconstrained, orientation=[-71, 132]);
```




```
> DEplot3d({L1,L2,L3}, {x(t),y(t),z(t)}, t=0..50, stepsize=.015,
[[x(0)=.1,y(0)=1,z(0)=1]], scene=[x(t),y(t),z(t)], axes=none, shading=NONE,
style=line, thickness=0, scaling=unconstrained, orientation=[-70,-90]) ;
```



Use the shading and style options to change the colors and style of your Lorenz attractor to suit your viewing pleasure.

3.3 Experiment: The Lorenz Attractor

Explore the Lorenz attractor and verify its sensitivity to initial conditions and its attractive nature.

1. Following the example given in the section above, plot the Lorenz attractor. Create at least one point-plot (style=point) and one line-plot (style=line). Try changing the shading to create different colors. (Some types of shading are XYZ, XY, Z, ZGRAYSCALE, ZHUE, NONE.) Also try different values for your stepsize. Print your four favorite plots. Don't forget to click on the graphs with your mouse and move them around to different viewpoints! This is one of Maple's best features.

2. Show that the Lorenz attractor is indeed an attractor by choosing initial seeds "far" from $(x_0, y_0, z_0) = (0.1, 1, 1)$. Plot and print your results for at least three different initial seeds. Go out to at least a distance of 50 from the origin for one of your seeds, and farther if you want. Can you get far enough away such that the orbit of your initial seed is not eventually attracted back to the Lorenz attractor?

3. Show that the Lorenz attractor is sensitive to initial conditions. Using the initial seed from the example above, $(x_0, y_0, z_0) = (0.1, 1, 1)$, plot the Lorenz attractor for a *small* range of t. Then plot the Lorenz attractor for a few initial seeds near $(0.1, 1, 1)$—perhaps $(0.1, 1, 1.1)$ or $(0.1, 1, 1.000001)$. It will be advantageous to use a small stepsize, set style=line, set thickness=0, and also choose shading=none. When creating these plots a range of 10 seconds or so would do fine. Recall, however, from Chapter 2 that the smaller the difference between your initial seeds, the higher out in the orbit you will have to go (higher t-value) to see sensitivity. Recall also that we are only interested in the eventual behavior of the orbit, and so throwing out, say, the first 10 seconds and plotting $t = 10...20$ may not be a bad idea. You will probably have to play around a bit to get a good combination of parameters. Do your results indicate that the Lorenz attractor is sensitive to initial conditions? If needed, click on the graph with your mouse and rotate the attractor to a viewpoint that helps you make conclusions.

4. Do your results from parts 1, 2, and 3 above indicate that the Lorenz attractor satisfies the properties of a strange attractor? Write a *brief* essay and turn it in to your instructor along with your printed results from steps 1, 2, and 3 above.

3.4 Experiment: The Hénon Attractor

Explore the Hénon attractor and verify its fractal and attractive natures.

1. Following the example given in the section on the Hénon attractor above, plot the Hénon attractor. You don't need to plot 50,000 points. 20,000 or so should do just fine. Be sure to throw out the first few points. Print your results.

2. Show that the Hénon attractor is indeed an attractor by choosing initial seeds "far" from $(x_0, y_0) = (0, 0)$. "Far" here means up to about a distance of *one* from the

origin. If you choose initial seeds too far from the origin, Maple displays the statement "Floating Point Overflow. Please shorten axes." Why does this happen? Also, be sure *not* to throw out any points from the orbit. The transitive points are where we will see the attraction. Do this for three different initial seeds and print your results.

3. Show that the Hénon attractor is a fractal. Choose an area of the attractor to magnify *other than the one shown in the example above*. Create an second magnification and show that the second magnification looks like the first magnification. Depending on the power of your magnifications, you may need to plot more than 50,000 points to achieve higher detail—perhaps up to 200,000 or more points. Be forewarned that plotting 200,000 points may take Maple 10 minutes or more, depending on the speed of your computer. So choose your view and other `pointplot` options carefully or else you may have to go back and re-plot if you are not satisfied, taking another 10 minutes! Print your results.

4. Do your results from steps 1, 2, and 3 above indicate that the Hénon attractor satisfies the properties of a strange attractor? Why, do you think, were you not asked to graphically verify that the Hénon attractor shows sensitivity to initial conditions? What *technical* difficulty would we run into if we tried to verify this? (*HINT: Remember that the Hénon attractor is a map—i.e., a collection of single points.*) Write a brief essay discussing these questions and turn it in to your instructor along with your printed results from steps 1, 2, and 3 above.