UNIVERSITY of CALIFORNIA

SANTA CRUZ

INVESTIGATIONS OF THE CHAOTIC WATER WHEEL

A thesis submitted in partial satisfaction of the
requirements for the degree of

BACHELOR OF SCIENCE

in

PHYSICS

by

Davidson Laning

June 2011

The thesis of Davidson Laning is approved by:

_____   _____
Jason Nielsen   Adriane Steinacker
Advisor   Theses Coordinator

_____
David P. Belanger
Chair, Department of Physics

Abstract

INVESTIGATIONS OF THE CHAOTIC WATER WHEEL

by

Davidson Laning

The chaotic water wheel was studied for various parameter values. The setup for this system is given by a wheel with containers along its circumference to which water is added and allowed to leak. The equations for the wheel were obtained for an asymmetric distribution for the inflow of water, which also describe the system for a symmetric inflow as a special case. These equations were investigated both analytically and numerically. The system exhibited periodic and chaotic behavior, and the solutions converged to fixed points for certain parameter values. It was seen that the solutions had sensitive dependence on initial conditions in the chaotic regime. In this regime, the divergence of nearby trajectories was shown to be exponential. The exponential divergence was seen to be weaker for an asymmetric inflow of water than for a symmetric inflow. Period doubling was observed and a value for the Feigenbaum constant, δ, was estimated to be 4.43.

Contents

1

Introduction

Chaos is a phenomenon that can be observed in systems as diverse as a population model, a chemical reaction, and a pendulum. By studying the governing equations of these systems, a whole range of behavior can be observed by varying parameter values. We can take the damped and driven pendulum as an example. If all of the parameters such as damping and the angular frequency of the driving are held constant, and the driving force is varied, different behaviors will be exhibited. With no driving force, the pendulum will eventually settle into a state of no motion due to the damping. For a larger driving force, there will be periodic motion and even chaos [1]. The most complex of these regimes is the chaotic regime, and there are three important concepts needed to understand what this means. First, the behavior of the chaotic system is aperiodic, and this behavior persists. Second, the system is deterministic and there are no random processes occurring that affect the behavior. Third is the idea of sensitive dependence on initial conditions [2]. The first of these three means that there is no discernible pattern or repetition in the behavior and this behavior does not settle into any periodic motion. The second means that if we know

the initial conditions of the system, then we can predict what it is going to do. There is no coin tossing or dice rolling to determine the outcome. Instead, the behavior comes from the nonlinearity of the governing equations. The third means that two chaotic systems started simultaneously from extremely similar initial conditions will diverge. For example, if two chaotic double pendulums are released from slightly different initial positions, their positions at a later time will be completely different. This behavior is contrary to what an experimenter might intuitively expect: the two pendulums would stay in step for all time. The sensitivity described means that for chaotic systems, from a pendulum to the weather, we are unable to predict the future because the initial conditions cannot be measured to infinite accuracy [3].

The focus of this paper is the chaotic water wheel, for which various behaviors from simply sitting still to chaos will be observed and studied.

1.1   The Wheel

A basic setup for this system, shown in Figure 1.1, is a wheel with containers all along its circumference, reminiscent of a Ferris wheel. The downward pointing arrows at the top represent an inflow of water into the containers, and each container has a hole in the bottom from which this water can leak. The motion of the wheel is a result of the gravitational torque from the water in the containers plus damping torque.
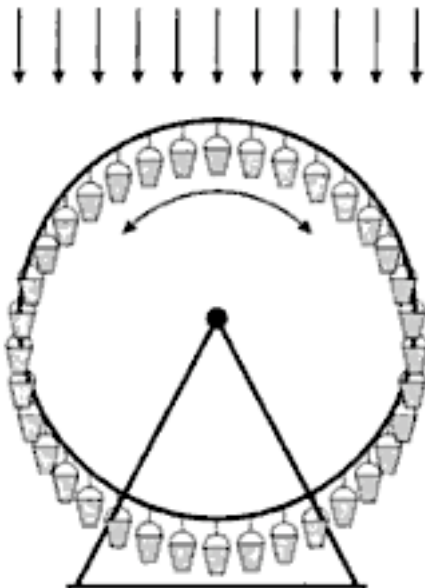
Figure 1.1: Setup for the water wheel. The downward arrows represent an inflow of water into the containers around the circumference of the wheel. This water is allowed to leak from holes in the containers and the wheel is allowed to rotate. The arrows in this figure represent show a symmetric inflow. The figure is from Tél and Gruiz [4].

1.2 Derivation of the Wheel Equations

For the derivation of the wheel equations we will follow the derivation for the symmetric case (water is added symmetrically to the containers) given in Strogatz [2].

The total mass in the wheel as a function of time is given by

$$M(t) = \int_{\theta_1}^{\theta_2} m(\theta, t) d\theta \tag{1.1}$$

where $m(\theta, t)$ is the mass at θ at time t and $\theta = 0$ is always at the top. ΔM, the change of mass for a section of the wheel as it is rotating, is given by

$$\Delta M = \left[\int_{\theta_1}^{\theta_2} Q(\theta) d\theta - \int_{\theta_1}^{\theta_2} Km(\theta, t) d\theta + m(\theta_1)\omega(t) - m(\theta_2)\omega(t) \right] \Delta t \tag{1.2}$$

where $Q(\theta)$ is the inflow of water, K is the constant leakage rate, and $\omega(t)$ is the angular velocity of the wheel. The first term in Equation (1.2) is from the inflow of water, the

second is due to the leakage, and the third and fourth are the mass of water carried into

and out of the range $\theta_2 - \theta_1$ as the wheel is rotating. We also have

$$m(\theta_1) - m(\theta_2) = -\int_{\theta_1}^{\theta_2} \frac{\partial m}{\partial \theta} d\theta \tag{1.3}$$

and

$$\frac{dM}{dt} = \int_{\theta_1}^{\theta_2} \frac{\partial m}{\partial t} d\theta. \tag{1.4}$$

By dividing both sides of Equation (1.2) by Δt, letting $\Delta t \to 0$, and using Equations (1.3)

and (1.4), we obtain

$$\frac{dM}{dt} = \int_{\theta_1}^{\theta_2} \left[Q(\theta) - Km(\theta, t) - \frac{\partial m}{\partial \theta} \omega(t) \right] d\theta = \int_{\theta_1}^{\theta_2} \frac{\partial m}{\partial t} d\theta. \tag{1.5}$$

Equation (1.5) is valid for all θ_1 and θ_2, so we can equate the integrands to obtain the

continuity equation:

$$Q(\theta) - Km(\theta, t) - \frac{\partial m}{\partial \theta} \omega(t) = \frac{\partial m}{\partial t}. \tag{1.6}$$

The torque balance equation for the wheel is given by

$$I\dot{\omega} = -\nu\omega + gr \int_0^{2\pi} m(\theta, t) \sin(\theta) d\theta, \tag{1.7}$$

where ν is the rotational damping rate, g is the gravitational constant, r is the radius of

the wheel, and I is the moment of inertia. The first term in Equation (1.7) is due to the

damping and the second term is from the torque due to gravity.

The inflow of water, $Q(\theta)$, and the mass, $m(\theta, t)$, are periodic functions and can

each be written as a Fourier series:

$$m(\theta, t) = \sum_{n=0}^{\infty} \left[a_n(t) sin(n\theta) + b_n(t) cos(n\theta) \right] \tag{1.8}$$

$$Q(\theta) = \sum_{n=0}^{\infty} \left[c_n sin(n\theta) + d_n cos(n\theta) \right]. \tag{1.9}$$

For a symmetric distribution for the inflow of water, all c_n will be zero. Keeping the c_n will give rise to an asymmetry parameter ρ in the final wheel equations. Partial derivatives of m with respect to t and θ are given by:

$$\frac{\partial m(\theta, t)}{\partial t} = \sum_{n=0}^{\infty} \left[\dot{a}_n(t) \sin(n\theta) + \dot{b}_n(t) \cos(n\theta) \right] \tag{1.10}$$

$$\frac{\partial m(\theta, t)}{\partial \theta} = \sum_{n=0}^{\infty} \left[na_n(t) \cos(n\theta) - nb_n(t) \sin(n\theta) \right]. \tag{1.11}$$

Plugging Equations (1.8) - (1.11) into (1.6) gives

$$\sum_{n=0}^{\infty} \left[\dot{a}_n(t) \sin(n\theta) + \dot{b}_n(t) \cos(n\theta) \right]$$

$$= \sum_{n=0}^{\infty} \left[c_n \sin(n\theta) + d_n \cos(n\theta) \right]$$

$$- K \sum_{n=0}^{\infty} \left[a_n(t) \sin(n\theta) + b_n(t) \cos(n\theta) \right]$$

$$- \omega \sum_{n=0}^{\infty} \left[na_n(t) \cos(n\theta) - nb_n(t) \sin(n\theta) \right]. \tag{1.12}$$

By equating the coefficients of sine and cosine terms on each side of the last equation, we obtain:

$$\dot{a}_n(t) = c_n - Ka_n(t) + \omega n b_n(t) \tag{1.13}$$

$$\dot{b}_n(t) = d_n - Kb_n(t) - \omega n a_n(t). \tag{1.14}$$

We can then plug the Fourier series for $m(\theta, t)$ into the torque balance equation (1.7) to obtain

$$I\dot{\omega} = -\nu\omega + gr \int_0^{2\pi} \sum_{n=0}^{\infty} \left[a_n(t)sin(n\theta) + b_n(t)cos(n\theta) \right] sin\theta d\theta$$

$$= -\nu\omega + gr \int_0^{2\pi} a_1(t)sin^2\theta d\theta$$

$$= -\nu\omega + \pi gra_1(t). \tag{1.15}$$

This gives, along with Equations (1.13) and (1.14) with $n = 1$, a set of first-order nonlinear ordinary differential equations for the water wheel:

$$\dot{\omega} = [-\nu\omega + \pi gr a_1(t)]/I$$

$$\dot{a}_1(t) = c_1 - K a_1(t) + \omega b_1(t)$$

$$\dot{b}_1(t) = d_1 - K b_1(t) - \omega a_1(t). \tag{1.16}$$

We are now going to make a change of variables, given below, to morph these equations into a more familiar form.

$$b_1 = -\frac{\nu K}{\pi gr} z + \frac{d_1}{K}$$

$$a_1 = \frac{\nu K}{\pi gr} y + \frac{c_1}{K}$$

$$\omega = Kx$$

$$t = \frac{\tau}{K}. \tag{1.17}$$

Equations (1.18) show the final form of the water wheel equations:

$$\dot{x} = \sigma(y - x + \rho)$$

$$\dot{y} = r_R x - xz - y$$

$$\dot{z} = xy - z + \rho x, \tag{1.18}$$

with $\dot{x} = \frac{dx}{d\tau}$, $\dot{y} = \frac{dy}{d\tau}$, $\dot{z} = \frac{dz}{d\tau}$, $\sigma = \frac{\nu}{IK}$, $\rho = \frac{\pi gr c_1}{K^2 \nu}$, $r_R = \frac{\pi gr d_1}{K^2 \nu}$, and σ, r_R, $\rho > 0$. These will be studied throughout the rest of this paper. The wheel equations for $\rho = 0$ are identical to the Lorenz equations, given below, for $b = 1$.

$$\dot{x} = \sigma(y - x)$$

$$\dot{y} = r_R x - xz - y$$

$$\dot{z} = xy - bz. \tag{1.19}$$

These equations were introduced by Edward Lorenz in a 1963 paper titled *Deterministic Nonperiodic Flow* as a simplification of a system used to study convection [5]. We now move on to the analytical study of the water wheel equations. From now on, the subscript R on the Rayleigh number, r_R, will be dropped.

2

Fixed Points and Bifurcations

The solutions for the fixed points of the wheel equations are to be addressed first. The numerical solutions of the water wheel equations can be plotted in (x, y, z) phase space to reveal the trajectories. When a trajectory reaches a fixed point of the system, it stays at the point for all time after. To obtain the fixed points for the water wheel, we can set Equations (1.18) equal to zero and solve for (x, y, z).

$$0 = y - x + \rho$$

$$0 = rx - xz - y$$

$$0 = xy - z + \rho x \tag{2.1}$$

Using equations (2.1), we can derive an equation given by (2.2), which is solved to give the x parts of the fixed points. We concentrate on x because this reveals how the wheel is rotating $(x = \omega/K)$.

$$x^3 - x(r - 1) - \rho = 0 \tag{2.2}$$

For $\rho = 0$, there is a fixed point $(x, y, z) = (0, 0, 0)$ for all r. For $r > 1$, there are two fixed points $(x, y, z) = (\pm\sqrt{r-1}, \pm\sqrt{r-1}, r-1)$. For nonzero ρ, the x value of the fixed points are obtained from Equation (2.2). Figure 2.1 shows four plots of the x part of the fixed point against r. For $\rho = 0$ there is always a fixed point at $x = 0$ corresponding to no rotation of the wheel. At $r = 1$ there is a pitchfork bifurcation where the single fixed point bifurcates into two stable fixed points corresponding to constant rotation and an unstable fixed point at $x = 0$. For nonzero ρ there is now an imperfect bifurcation which occurs at larger r for larger values of ρ. Also notice that there is no longer a fixed point at $x = 0$. Because the distribution of the inflow of water is asymmetric, the torque due to the water added to one side of the wheel cannot be perfectly balanced by the torque from the water added to the other side, and the wheel cannot sit perfectly still.

To determine where the imperfect bifurcations will occur, the value of ρ on the boundary where $x^3 - x(r-1) = \rho$ has one and three real solutions can be calculated. Figure 2.2 shows a plot of $x^3 - x(r-1)$ for $r = 8$ and different values of ρ. There will be a minimum and maximum as long as $r > 1$, and when ρ crosses a minimum or maximum as it is varied, there will be a change in the number of fixed points of the system. There is a local minimum at $x_{min} = \sqrt{r-1}$ and a local maximum at $x_{max} = -\sqrt{r-1}$. At these points the function has the values $\rho_c(r) = \pm\frac{2}{3}(r-1)\sqrt{\frac{r-1}{3}}$.

Figure 2.3 shows a plot of $\rho_c(r)$. As the parameters ρ and r are varied, a bifurcation will occur when one of the curves is crossed. Again we see that for larger values of ρ, bifurcations occur for larger values of r. Figure 2.4 shows a plot of the surface of solutions of Equation (2.2) over the (r, ρ) plane. This plot, known as a cusp catastrophe surface, shows where there are one and three fixed points for a range of r and ρ [4]. For example, for the
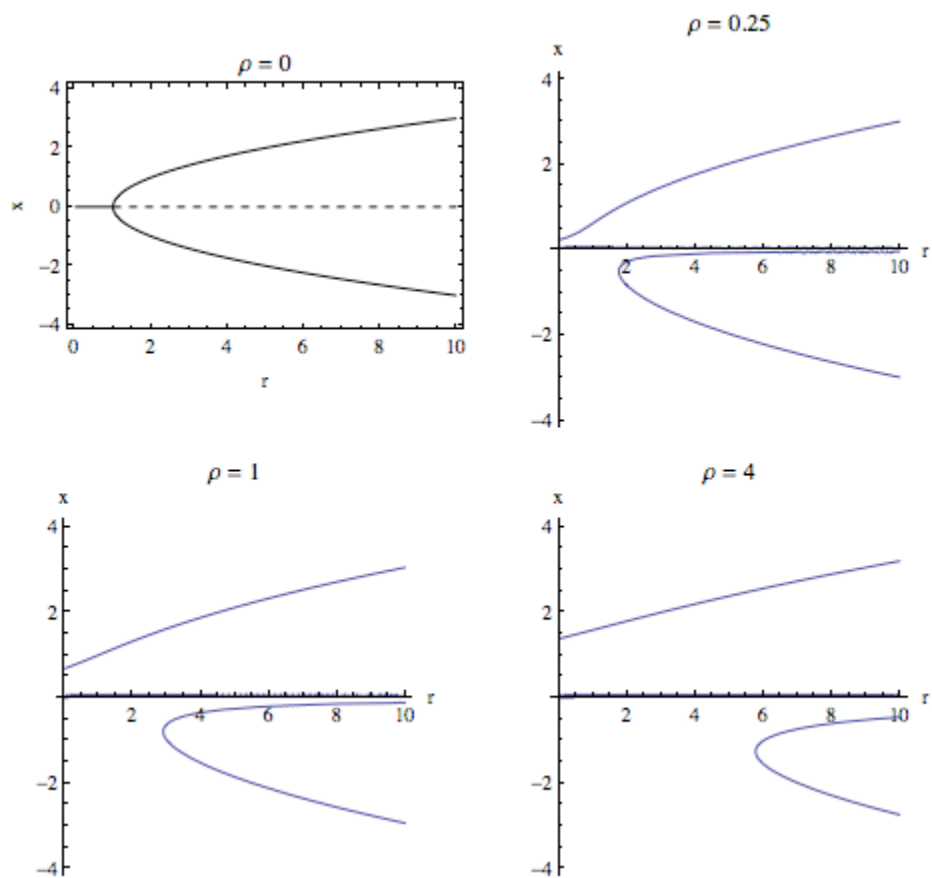
Figure 2.1: Bifurcation diagrams for varying ρ. For $\rho = 0$, a bifurcation occurs at $r = 1$. For nonzero ρ, there is now an imperfect bifurcation and there is no longer an unstable fixed point at $x = 0$.

parameter values $(r, \rho) = (9, 1)$, the figure shows there are three fixed points corresponding

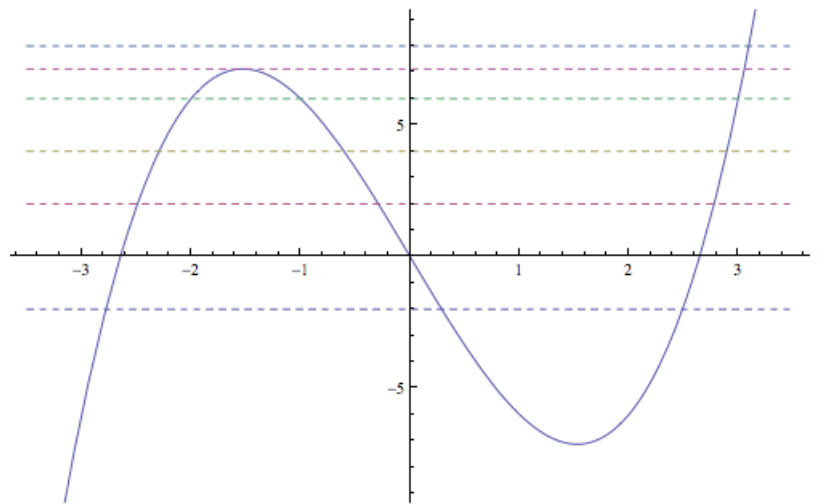to the three layers of the surface over the point.

Figure 2.2: Plot of $x^3 - x(r - 1)$ and ρ for $r = 8$ and various values of ρ. The solutions of Equation (2.2) are shown by the intersections of the curve with the dashed lines. As ρ is varied, a bifurcation occurs when the plot goes from showing one solution to three. This happens for this example when ρ is about 7.
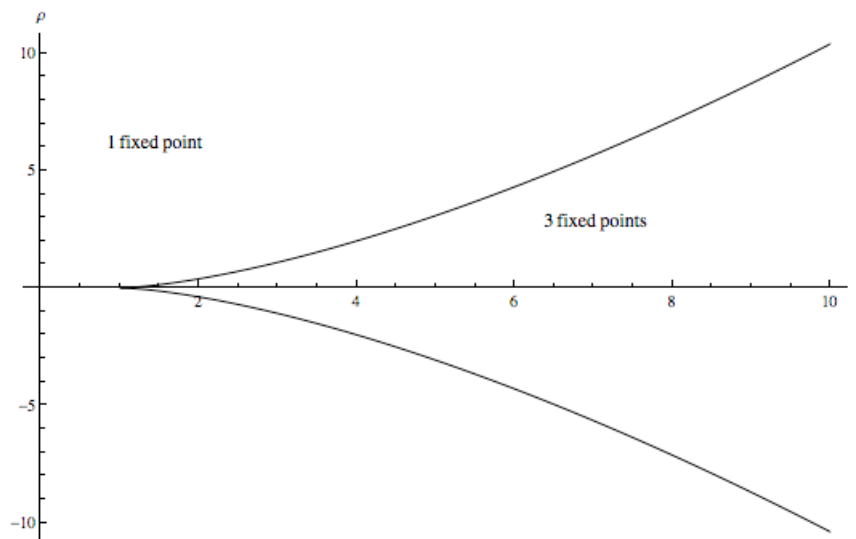

Figure 2.3: Stability diagram showing a plot of $\rho_c(r)$. The plot shows the two different regions of (r,ρ) space with a different number of fixed points. The plotted curve is the boundary where bifurcations occur as the parameters are varied.
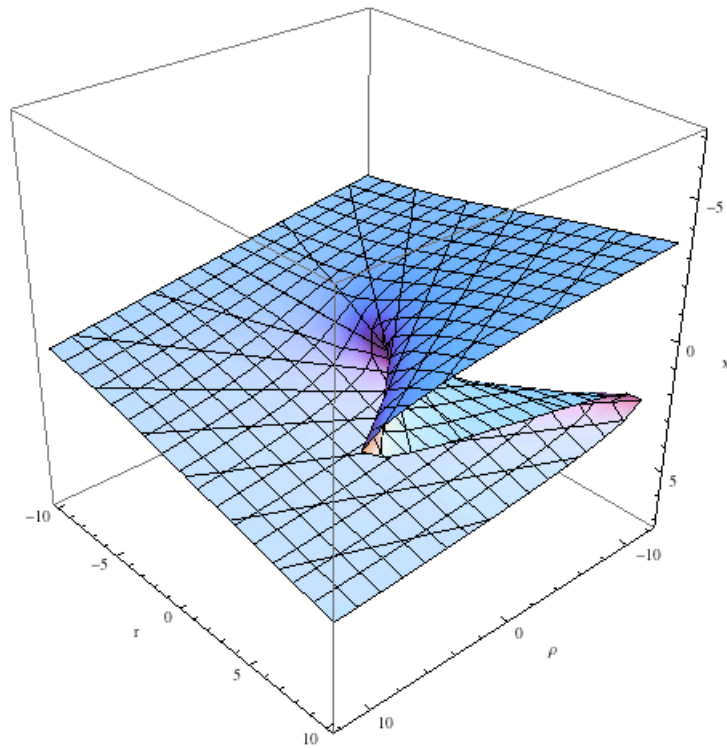
Figure 2.4: Cusp catastrophe surface [4]. For the region of the (r, ρ) plane where three fixed points exist, the surface has three layers.

3

Numerical Solutions for $\rho = 0$

Mathematica[1] will be used to numerically solve the water wheel equations and create diagrams to shed light on the behavior of the wheel.

The water wheel will first be studied for $\rho = 0$, the symmetric case. Some effects of nonzero ρ will be investigated later on. For the rest of this paper, σ will be set to 10.

Figure 3.1 shows four plots of x versus t for various values of r. For all of these plots, the initial conditions are $(x[0], y[0], z[0]) = (1, 1, 1)$. In the first plot, where $r < 1$, it is seen that x goes to zero, the only existing fixed point. For Rayleigh number $r = 13$, where there are now two stable fixed points and one unstable point at the origin, x settles down to a negative value, corresponding to constant rotation with negative angular velocity. For $r = 27.65$ it appears that there is a new type of motion and after some transient behavior, x becomes periodic. For $r = 29$ there appears to be no settling into either a fixed point or periodic motion.

Figure 3.2 shows four plots for the same values of r as Figure 3.1, but this time

---

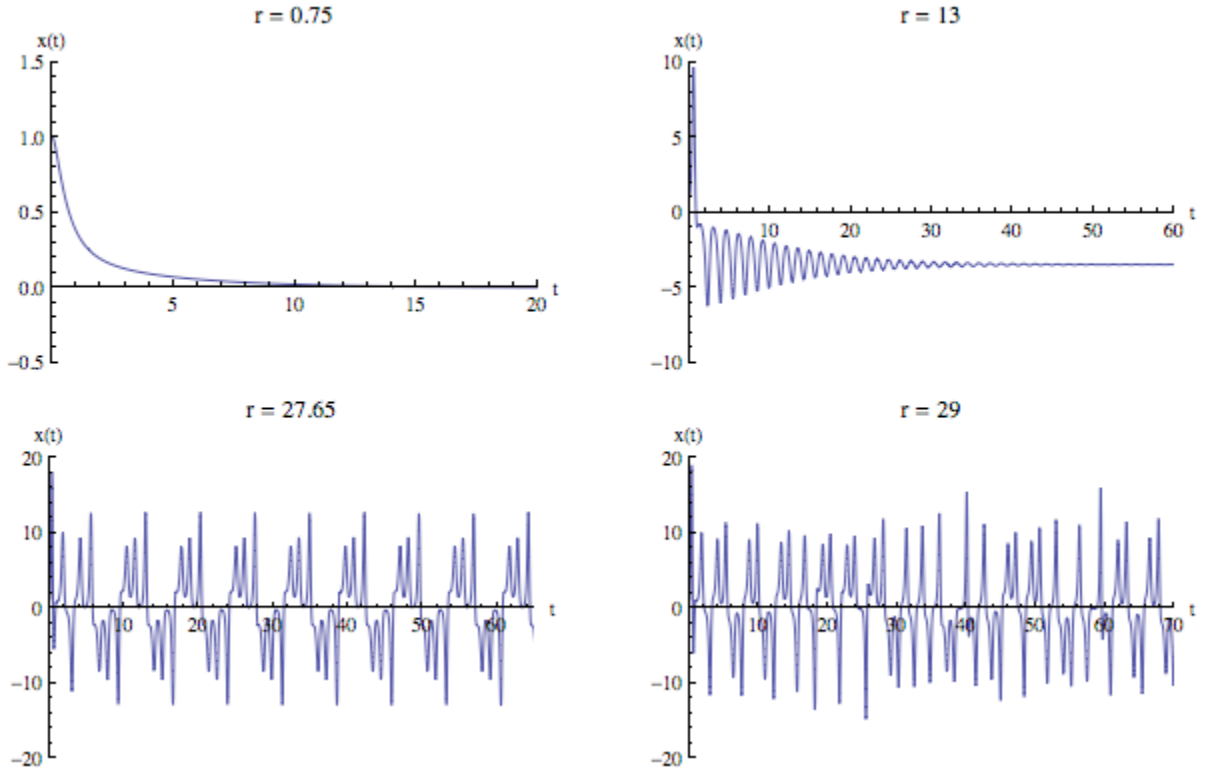[1]Mathematica 8 For Students was used for all of the plots in this paper.

Figure 3.1: Plots of $x(t)$ versus t for different values of r. The initial conditions are (x, y, z) $= (1, 1, 1)$. For $r = 0.75$, the wheel stops rotating. For $r = 13$, after some initial transients, the wheel rotates with constant angular velocity. For $r = 27.65$, the wheel exhibits periodic motion after the initial transients are gone. In the plot for $r = 29$, there appears to be no periodic motion or settling into a fixed point.

the solutions are plotted in three dimensional phase space. The initial conditions have been changed to (x[0],y[0],z[0])=(-3,4,20). In the first plot, with Rayleigh number $r = 0.75$, the trajectory approaches the fixed point at $(0, 0, 0)$, the same as before. For $r = 13$, the trajectory goes into a spiral toward the fixed point with positive x, corresponding to steady rotation with positive angular velocity. For $r = 27.65$ and $r = 29$, the the trajectories are plotted after some time has passed to omit the transient behavior. As seen before, there is periodic and chaotic behavior.

Figure 3.3 shows plots of four trajectories, started at different points, for $r = 29$.

The three dots show trajectories started at the fixed points of the system. One is at the origin and the other two, corresponding to steady angular velocity, are shown in red. The trajectories started at these initial conditions will remain at the same points. The most interesting trajectory, shown in blue, starts at the top of the plot, is attracted downward, and traces out a strange attractor.

3.1   Sensitive Dependence

To investigate the dependence on initial conditions, two trajectories will be started extremely close in phase space and their x values will be plotted against time. This is shown in Figure 3.4. For both trajectories, the initial y and z will be equal to 1 and the initial x values will be 1 and 1.00000001. In the beginning, both plots overlap, but at around $t = 40$ it is seen that they become very different. This behavior illustrates the sensitive dependence of initial conditions for chaotic systems. In order to see that the divergence is exponential, we can plot the natural logarithm of the distance between two trajectories in phase space, $\ln\left[\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}\right]$, against time. Figure 3.5 gives such a plot for this system, for $r = 29$. At $t = 0$, we see that the distance is extremely small, but as time goes on $\ln[d(t)]$ grows roughly linearly. Eventually the curve levels off as the two trajectories are confined to the strange attractor. Trajectories are attracted to this strange attractor and trajectories on the attractor diverge. The slope of the roughly linearly increasing region gives an estimate for the largest Liapunov exponent [1]. A positive exponent indicates exponential divergence, and this divergence is seen for the parameter values used for Figure 3.5.

3.2 Bifurcation Diagram

Up until now, behavior has been shown for only a few different values of r. To see what happens for a large range of r, we can construct a bifurcation diagram numerically. For small values of r, this should look like the analytical solution given in Figure 2.1. For larger values of r, as we have seen, there exist more than just fixed points. Figure 3.6 gives the bifurcation diagram for the symmetric case. It is made by integrating the wheel equations, throwing out transients, and then plotting values of x in steps of 0.1. The value of r is then increased by one step and the process occurs again. The plot shows the resulting values of x for three different trajectories. For smaller values of r, the plot matches that of Figure 2.1 for $\rho = 0$. The stable fixed points have been plotted over in red and the dashed purple line shows the unstable fixed point at the origin. At r just below 15, the plot changes dramatically and we see that the trajectories no longer settle down to a fixed point. It is also seen that the larger the value of r, the larger the range of x values. There is also a section between $r = 25$ and $r = 30$ that looks different. Remember that there was limit cycle behavior for $r = 27.65$. This region will be explored more shortly, but first there is a discussion of Lorenz maps.

3.3 Lorenz Maps and Orbit Diagrams

Figure 3.7 shows a plot of z against time for $r = 29$. To look at the relationship between a local maximum of $z(t)$ and the next maximum, we can create what is known as a Lorenz map [4]. To do this, the wheel equations are solved numerically and the values of z for the maximums are kept track of after transients are discarded. The value of the

maximum, z_n, is then plotted against the value of the next maximum, z_{n+1}. This is shown

for $r = 29$ in Figure 3.8 . We see there is a clear relationship between the maximums.

With this tool, we can look at the interesting region of the bifurcation diagram

between $r = 26$ and $r = 28$. Two Lorenz maps are shown in Figure 3.9 for $r = 26.95$ and

27.65. The map on the right has only three blobs, meaning the maximum is the same, or

very close to the same, after going around the attractor three times. The plot points to

limit cycle behavior. As r is decreased, it appears there is period doubling, which occurs

between $r = 27.65$ and $r = 26.95$.

The period doubling can be seen in a plot of z_{max} for a range of r. This is shown

in Figure 3.10. We can see that for a range of r values, the maxima only take certain values.

We can also see clearly that the top branch splits as r is decreased. Figure 3.11 zooms in on

this area, giving a better view of the period doublings. This period doubling is a feature of

many chaotic systems, and has been observed in physical experiments [1]. From the figure

we can see that as r is decreased at a constant rate, the doublings occur more rapidly. In

other words, the range of r needed to be covered before the next period doubling occurs

decreases. The Feigenbaum constant, given by

$$\delta = \lim_{n \to \infty} \frac{r_n - r_{n-1}}{r_{n+1} - r_n} = 4.669...,$$

tells us approximately where the next period doubling will occur as a parameter r is varied

for the chaotic system [4]. Figure 3.11 can be used to obtain a rough estimate of this constant

for the wheel system. Using the values $r = 27.19$, 26.88, and 26.81 for the locations of the

period doublings, the estimate for δ comes out to 4.43. Figure 3.12 shows another plot of

z_{max} against r for a larger range of r. In this figure we can see two regions with limit cycle

behavior. As r is increased, there is period halving in the gap in the region $26 < r < 28$

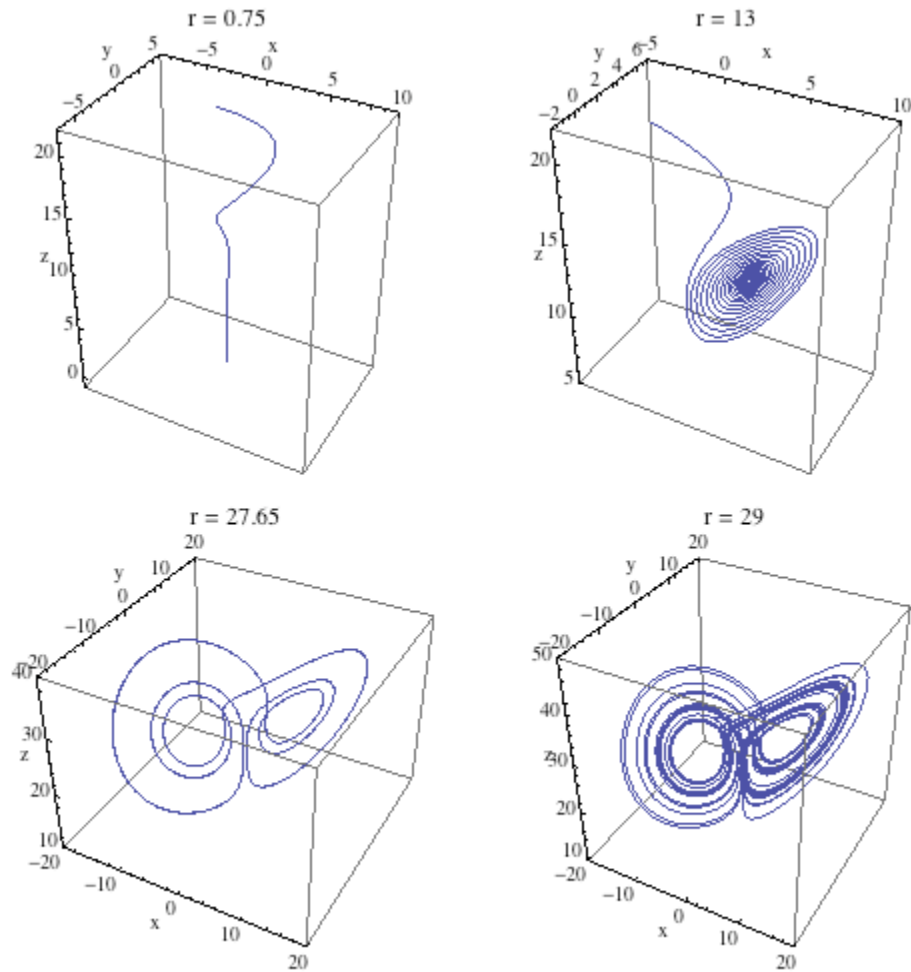and period doubling in the gap in the region $39 < r < 43$.

Figure 3.2: 3D phase space plots for different values of r with initial conditions (x, y, z) $= (-3, 4, 20)$. For $r = 0.75$, the trajectory starts at the top of the plot and goes into the stable fixed point at the bottom. For $r = 13$, the trajectory spirals into the fixed point corresponding to positive constant angular velocity. The plots for $r = 27.65$ and $r = 29$ are shown with the transient behavior omitted where a limit cycle and a chaotic attractor are visible.
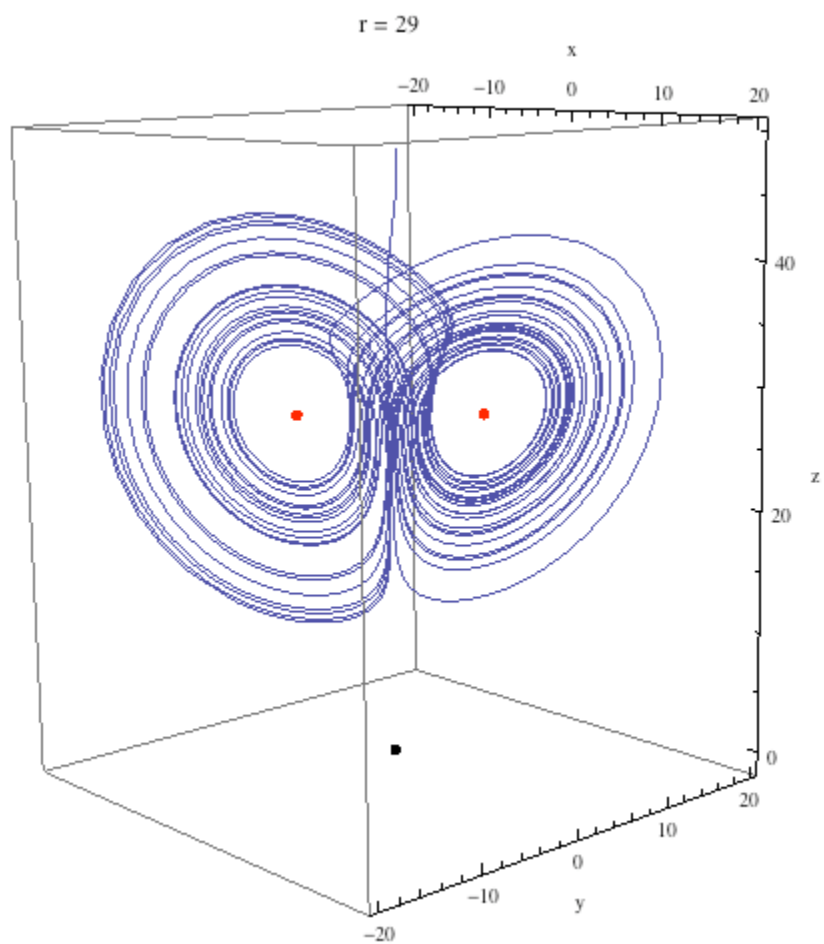
r = 29

Figure 3.3: 3D phase space plot showing four trajectories for $r = 29$. The black and red blobs show trajectories started at the fixed points of the system. Once started at the fixed points, the trajectories remain at the fixed points. The last trajectory starts at the top of the plot, at $(x, y, z) = (0, 1, 49)$, and traces out a strange attractor after transient behavior.
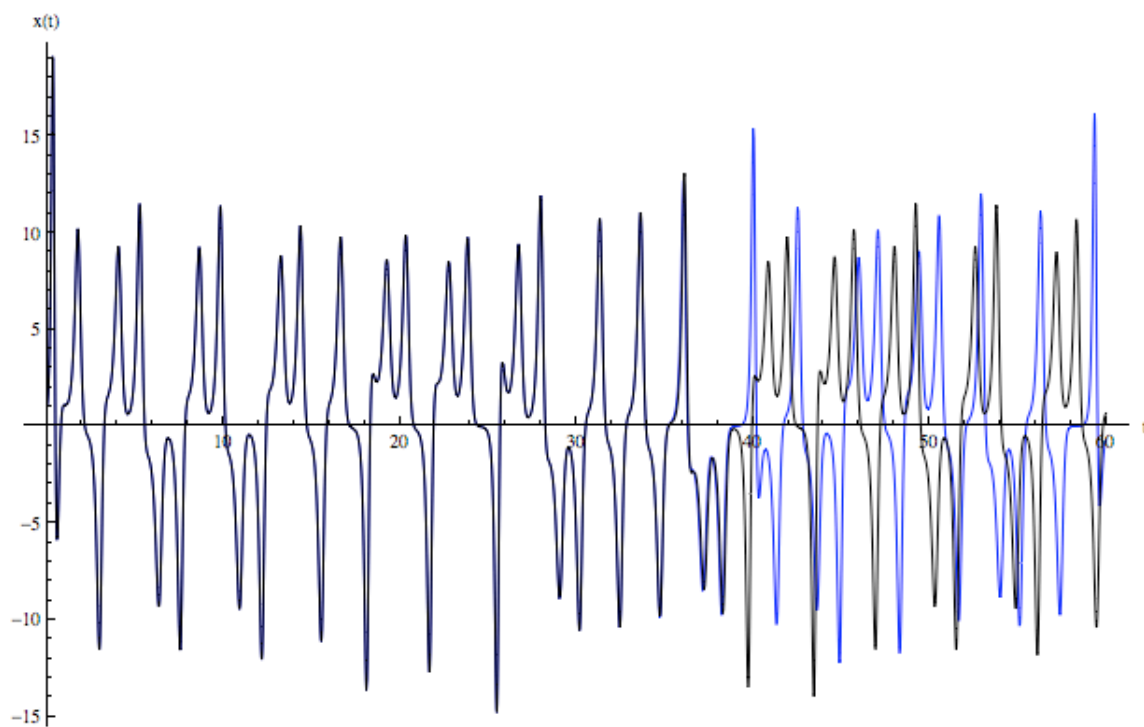
Figure 3.4: Plot of $x(t)$ versus t for two close sets of initial conditions. The initial x values are 1 and 1.0000001 for the blue and black plots, respectively. The initial y and z values are 1 for both plots. We can see that even for extremely close initial conditions, the two plots end up diverging at around $t = 40$.
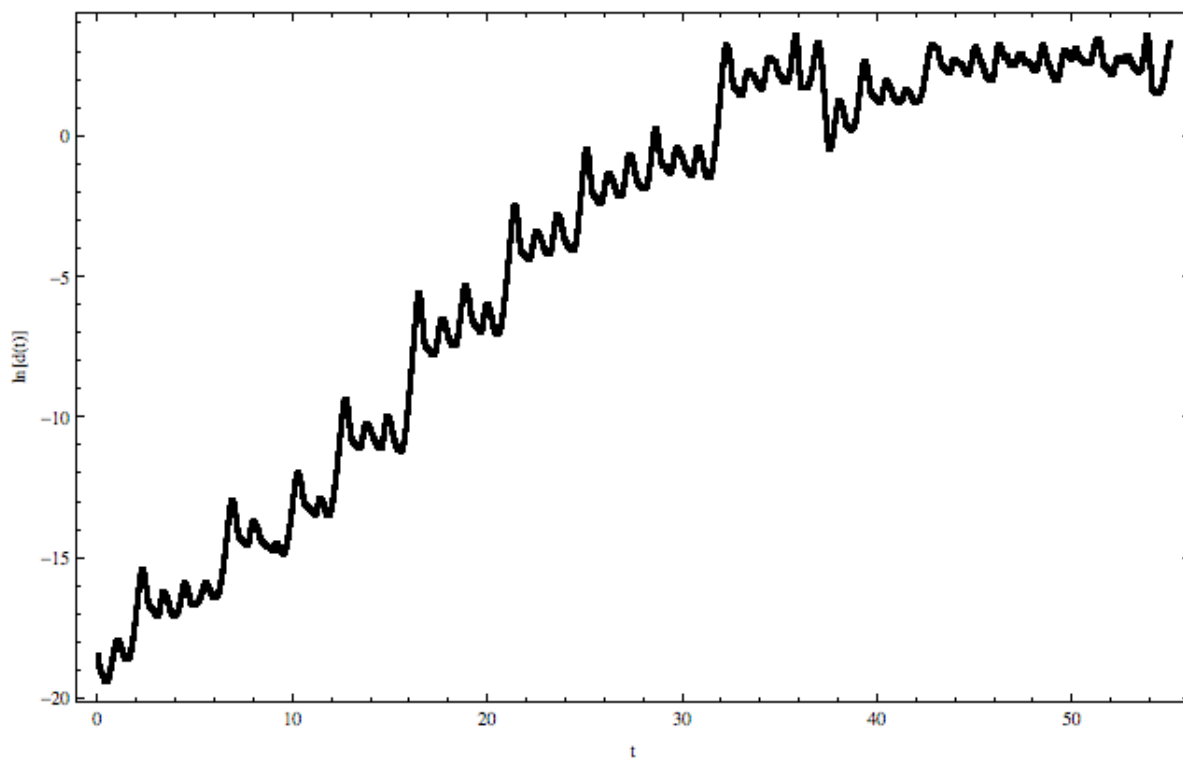
Figure 3.5: Plot of $\ln[d(t)]$ versus t for two trajectories started close together with $r = 29$. The plot increases roughly linearly, indicating exponential divergence, until it levels off. This leveling off is due to the finite size of the strange attractor. The two trajectories do not leave the attractor so there is a maximum distance they can be from each other.
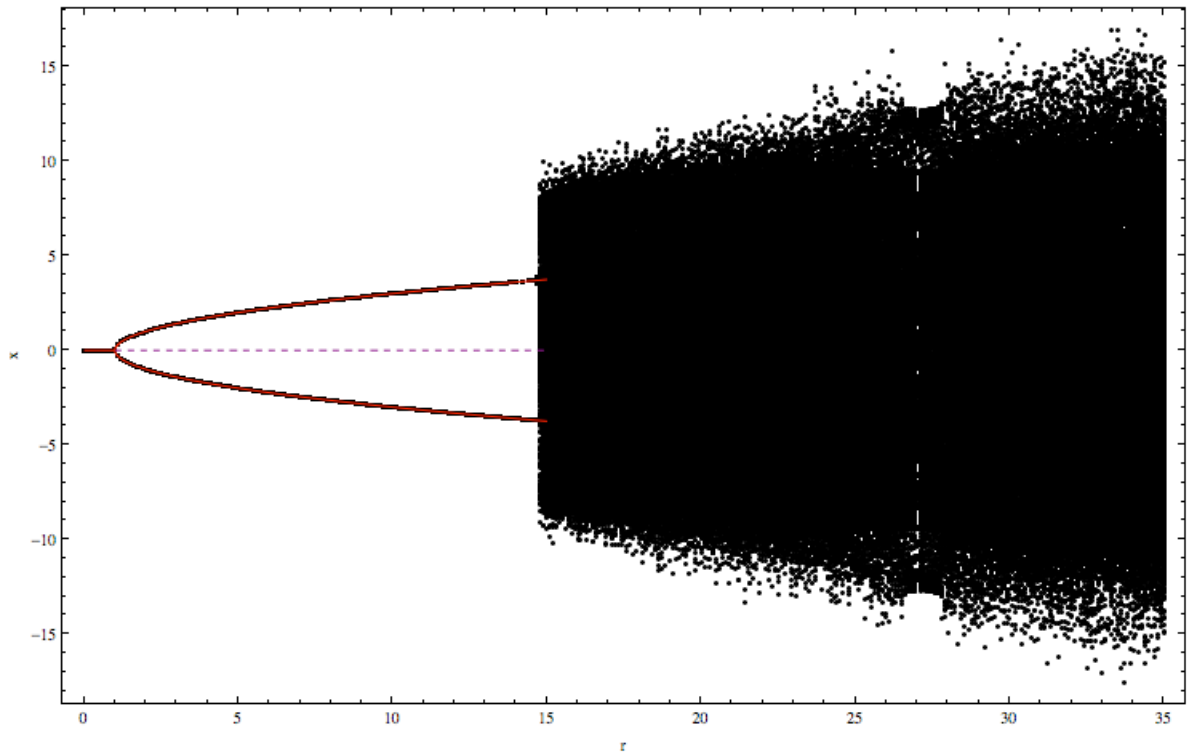
Figure 3.6: Bifurcation diagram for $\rho = 0$. Up until just below $r = 15$, the numerical bifurcation diagram agrees with the analytical diagram, plotted in red. The purple dashed line shows the unstable fixed point for $x = 0$. For r above some value, there is a dramatic change and x varies over a large range.
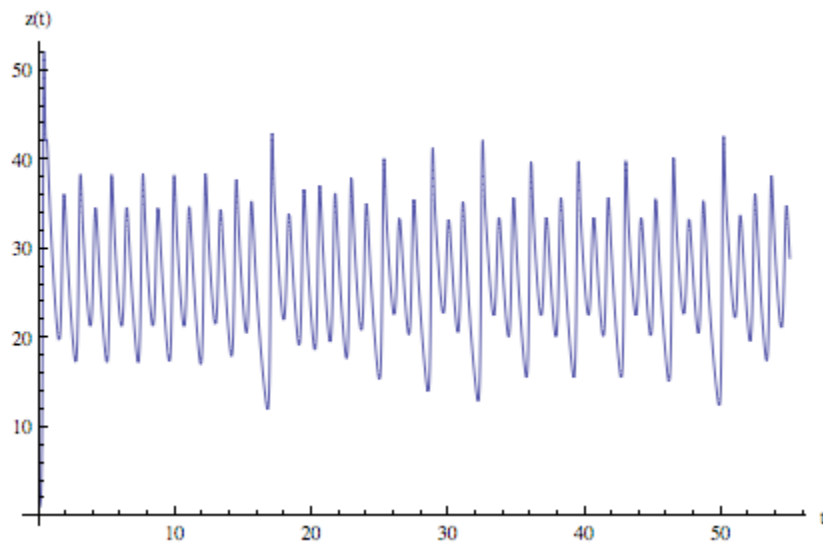

Figure 3.7: Plot of z against time for $r = 29$. A Lorenz map shows the relation ship of one maximum of this plot to the next maximum.
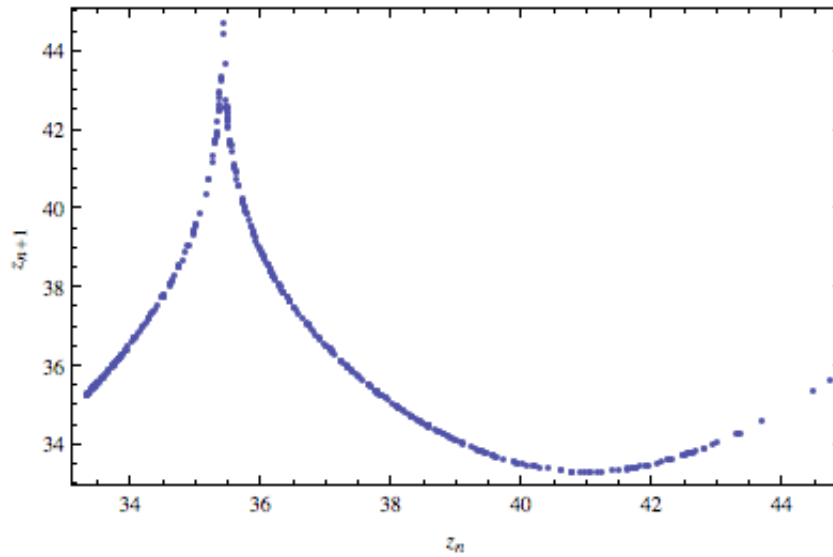
Figure 3.8: Lorenz Map for $r = 29$. This plot shows the relationship of one maximum of z versus t to the next. Given the value of a maximum z_n, we can predict the value of the next maximum, z_{n+1}
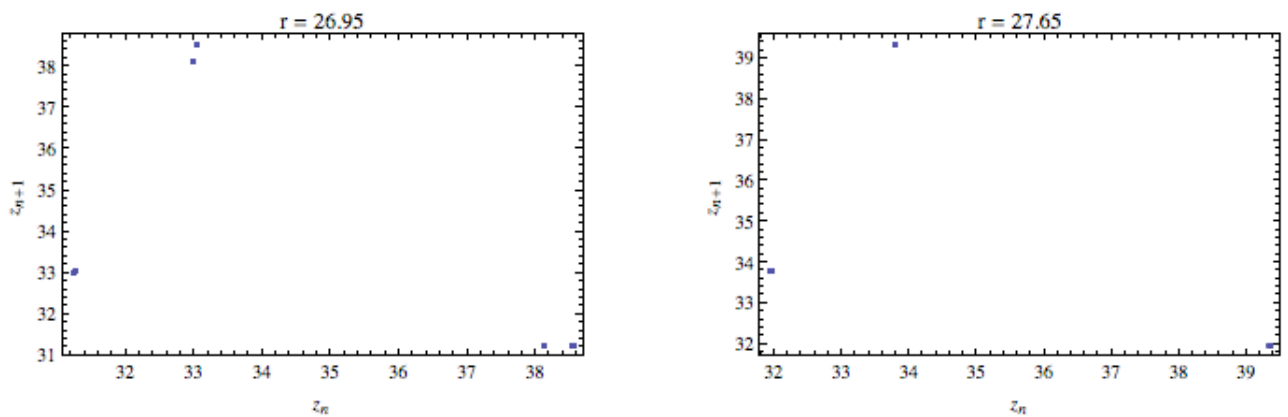
Figure 3.9: Lorenz Maps. These maps indicate a region of limit cycle behavior. As r is increased, there is a merging of the blobs, indicating period halving of the limit cycle.
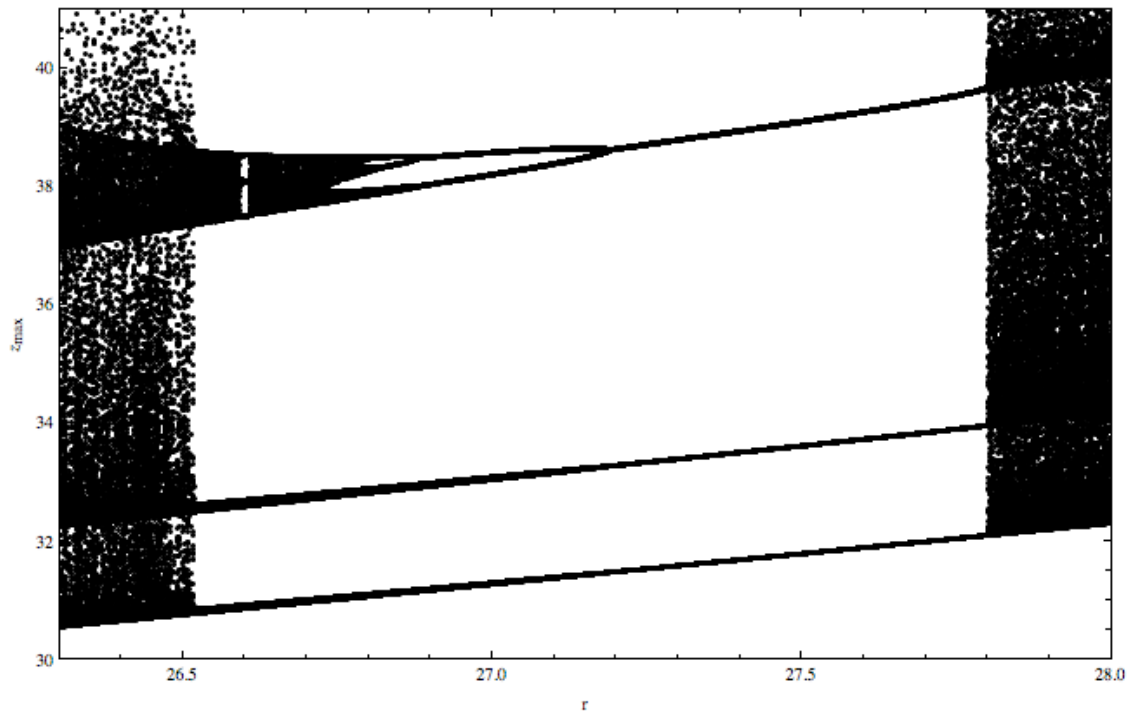
Figure 3.10: Z orbit diagram. This plot illustrates the period doubling behavior of limit cycles as r is decreased.
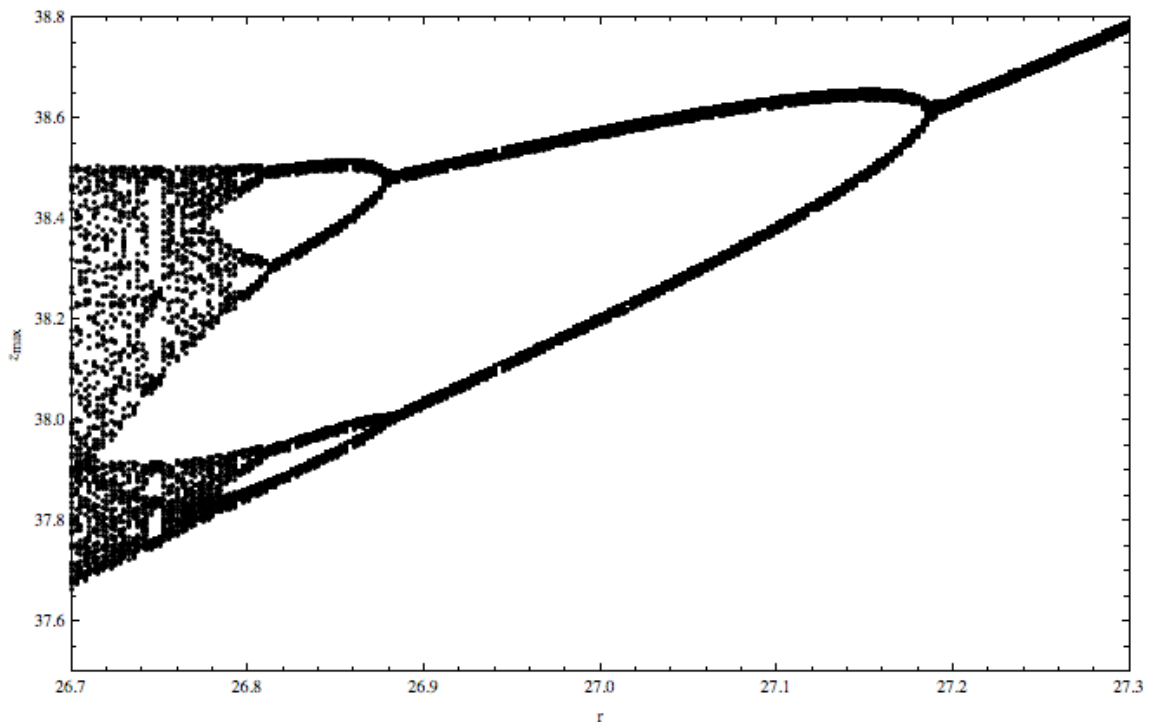
Figure 3.11: Zoomed in z orbit diagram. This plot zooms into the region of Figure 3.10 showing limit cycle period doubling. This plot can be used to obtain an estimate of the Feigenbaum constant, δ, described in Section 3.3.
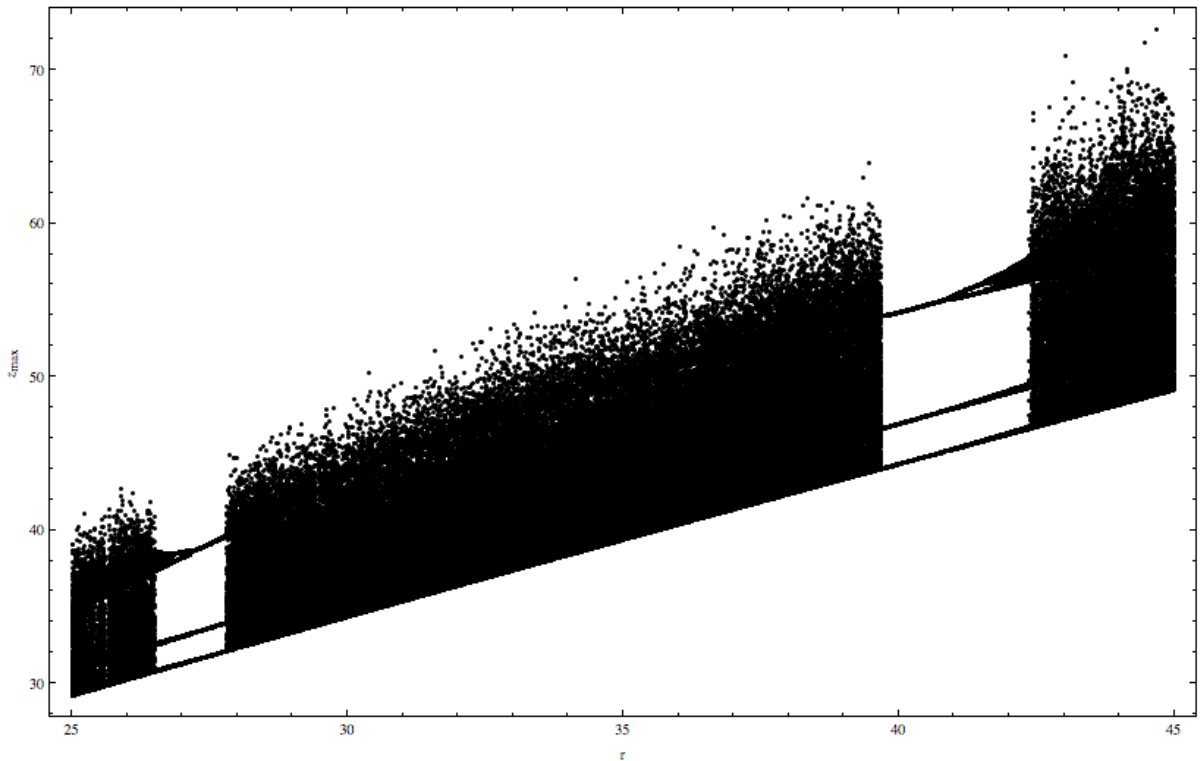
Figure 3.12: Z orbit diagram for a larger range of r. The region investigated earlier, with $r < 28$, is shown on the left side of the plot. The plot shows both period halving ($r \approx 27$) and period doubling ($r \approx 41$) as r is increased.

4

Numerical Solutions for $\rho \neq 0$

We now begin the investigation of the water wheel equations for nonzero ρ with numerical solutions. Having a nonzero ρ corresponds to an asymmetric inflow of water, and we will see this change in the solutions. Figure 4.1 shows four plots for $r = 29$ and increasing values of ρ. All plots were started with initial conditions $(1, 1, 1)$. It is seen that the wheel spends more time with a positive angular velocity, caused by the positive value of ρ. For $\rho < 0$, the wheel spends more time with negative angular velocity. As ρ is increased, the asymmetry grows until, as plot (d) shows, the wheel settles to a fixed point. There seems to still be chaotic behavior based on the appearance of plots (a) and (b). To see that there is still sensitive dependence on initial conditions, a plot similar to Figure 3.4 is shown in Figure 4.2 with $\rho = 4$ and $r = 29$. The initial conditions for the blue and black solutions are $(1, 1, 1)$ and $(1.0000001, 1, 1)$, respectively. The two plots overlap for some time, but at around $t = 60$ there is visible divergence. Figure 4.3 shows a plot of the natural logarithm of the distance of two trajectories in phase space for $\rho = 4$ and $r = 29$. We can see that there is still exponential divergence, but it is not as strong as for the $\rho = 0$ case. This is

(a) $\rho = .7$

x(t)

(b) $\rho = 2$

x(t)

(c) $\rho = 4$

x(t)

(d) $\rho = 7$

x(t)
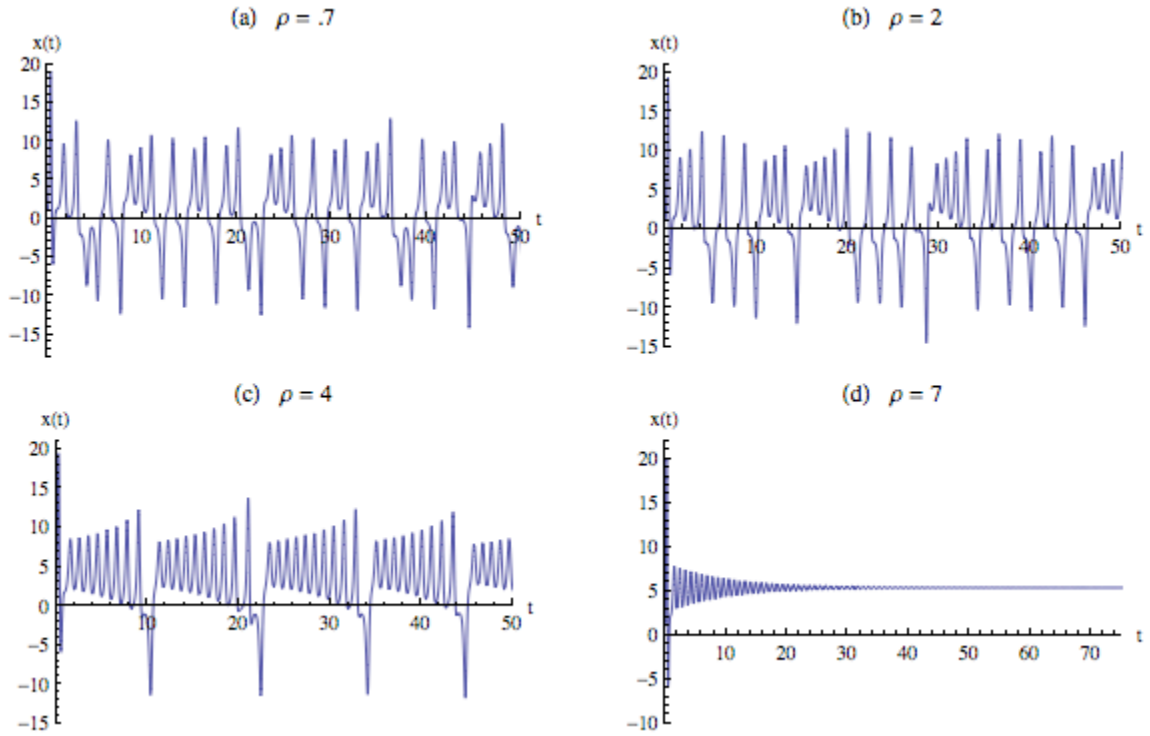
Figure 4.1: Four plots of x against t for $\rho \neq 0$, $r = 29$. For increasing values of ρ, the wheel spends more time with positive angular velocity until it settles into constant rotation shown in (d).

evident from the smaller slope of the roughly linear increasing region of the plot. Figures 4.4 and 4.5 show two bifurcation diagrams for $\rho = 0.1$ and 4, respectively with $r = 29$. For $\rho = 0.1$ we see the bifurcation diagram looks similar to Figure 3.6 for $\rho = 0$. The solution to Equation 2.2 for the fixed points is plotted in blue. Figure 4.5, with the larger ρ, shows the asymmetry more dramatically.

Finally, Figure 4.6 shows a plot of the maxima of z for a range of r. It shows numerous ranges of r for which there is limit cycle behavior.
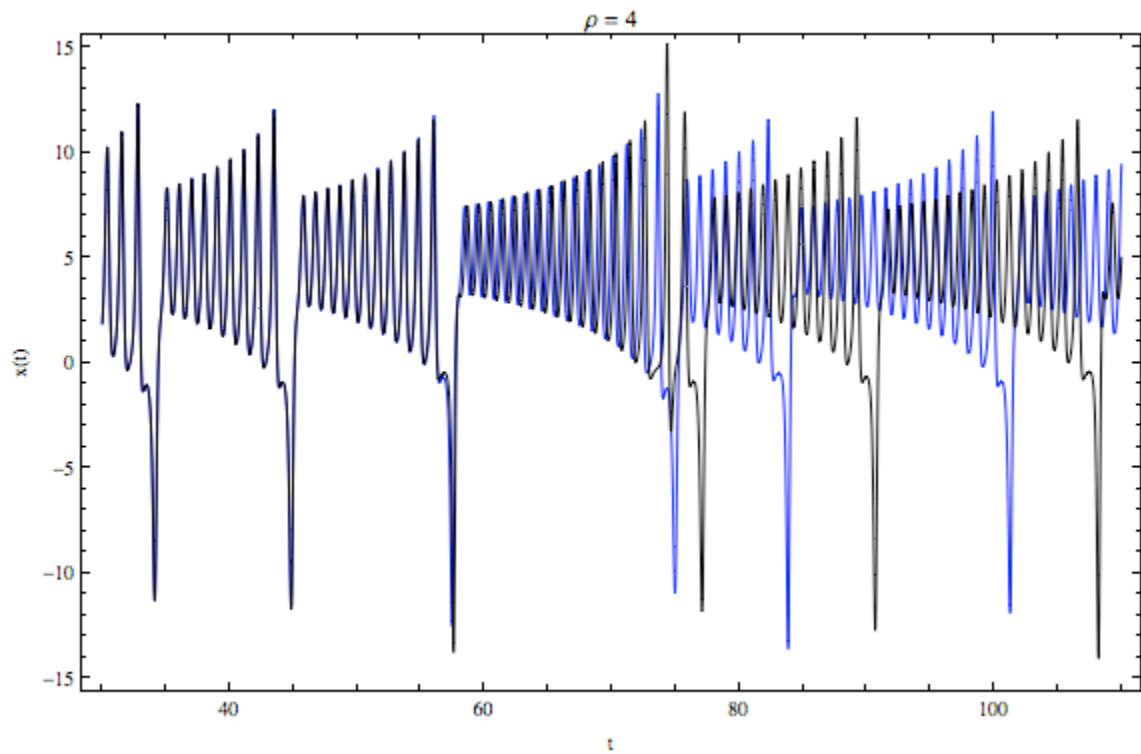
Figure 4.2: Two trajectories with similar initial conditions $(1, 1, 1)$, shown in blue, and $(1.00000001, 1, 1)$, shown in black. The divergence of these trajectories is clearly visible.
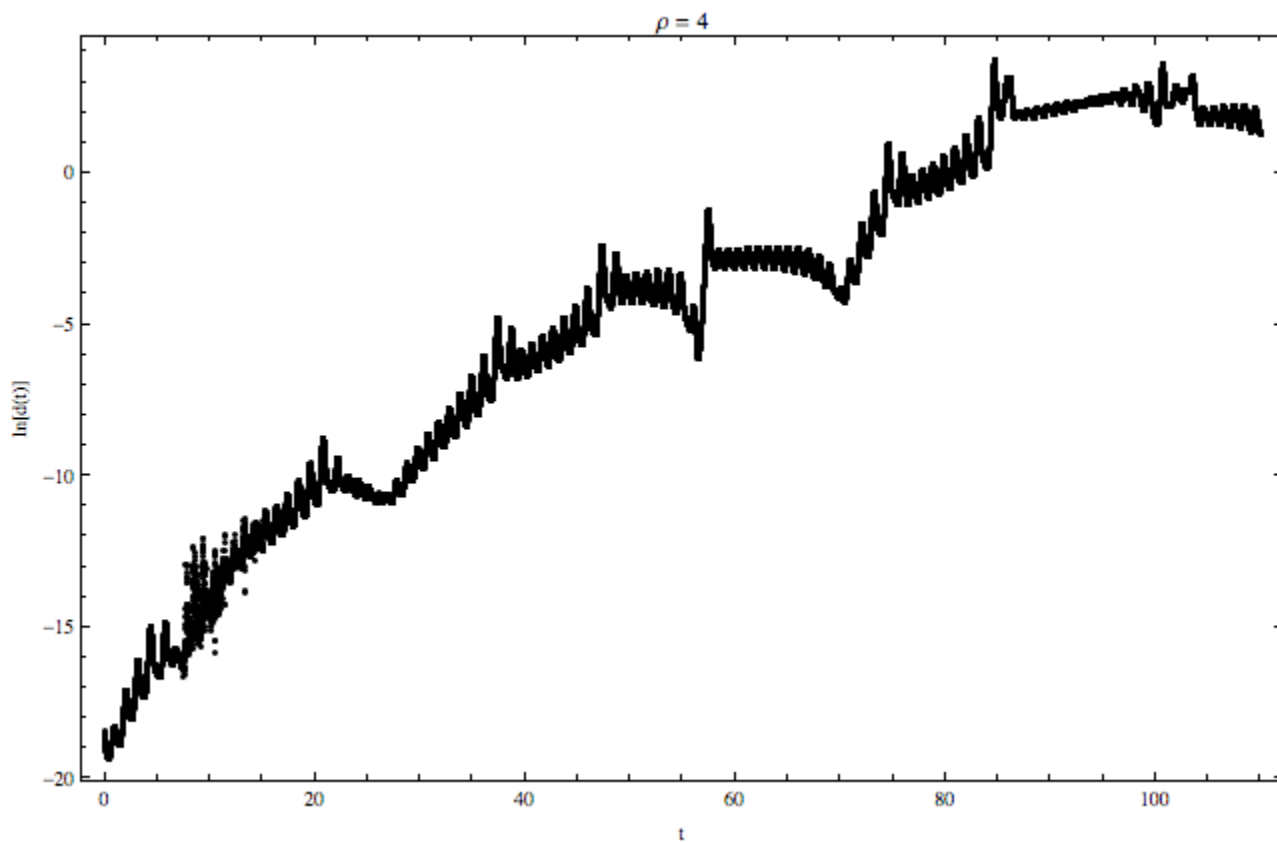
Figure 4.3: Plot of $\ln\left[d(t)\right]$ against t for two trajectories started close together for $\rho = 4$ and $r = 29$. The slope of the roughly linearly increasing region is less than that of the plot for the $\rho = 0$ case and means that the divergence is not as strong.
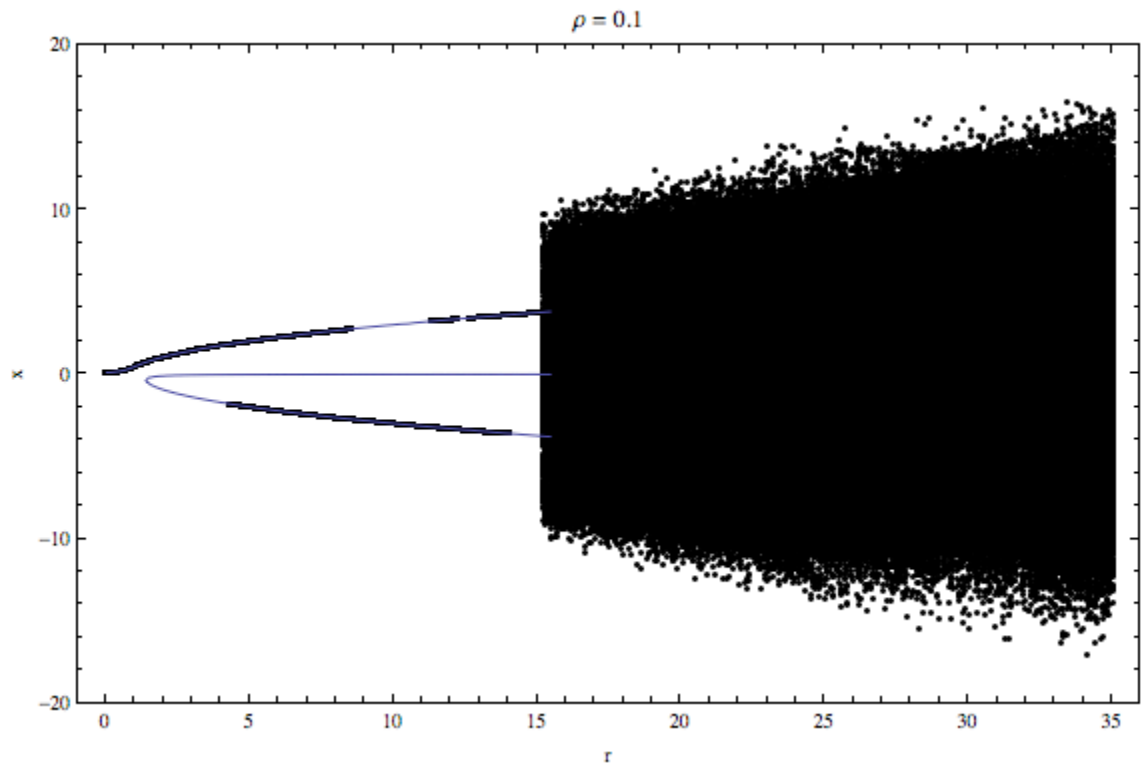
Figure 4.4: Bifurcation diagram for $\rho = 0.1$. The plot is similar to the $\rho = 0$ case, but the imperfect bifurcation is visible at $r \approx 1$.
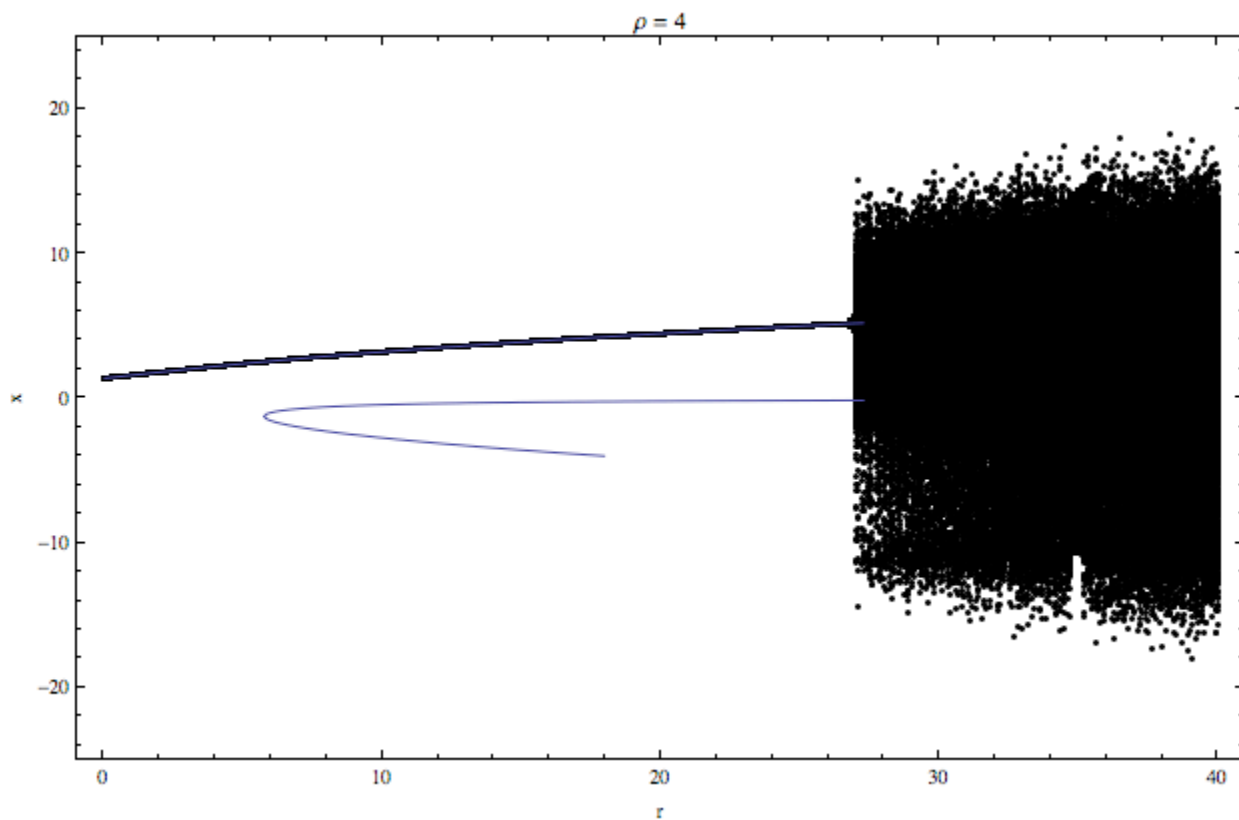
Figure 4.5: Bifurcation diagram for $\rho = 4$. This plot shows more dramatically the asymmetry for nonzero ρ than Figure 4.4 because of the larger value of ρ. For the initial conditions used, none of the trajectories settled to the fixed points corresponding to negative constant angular velocity.
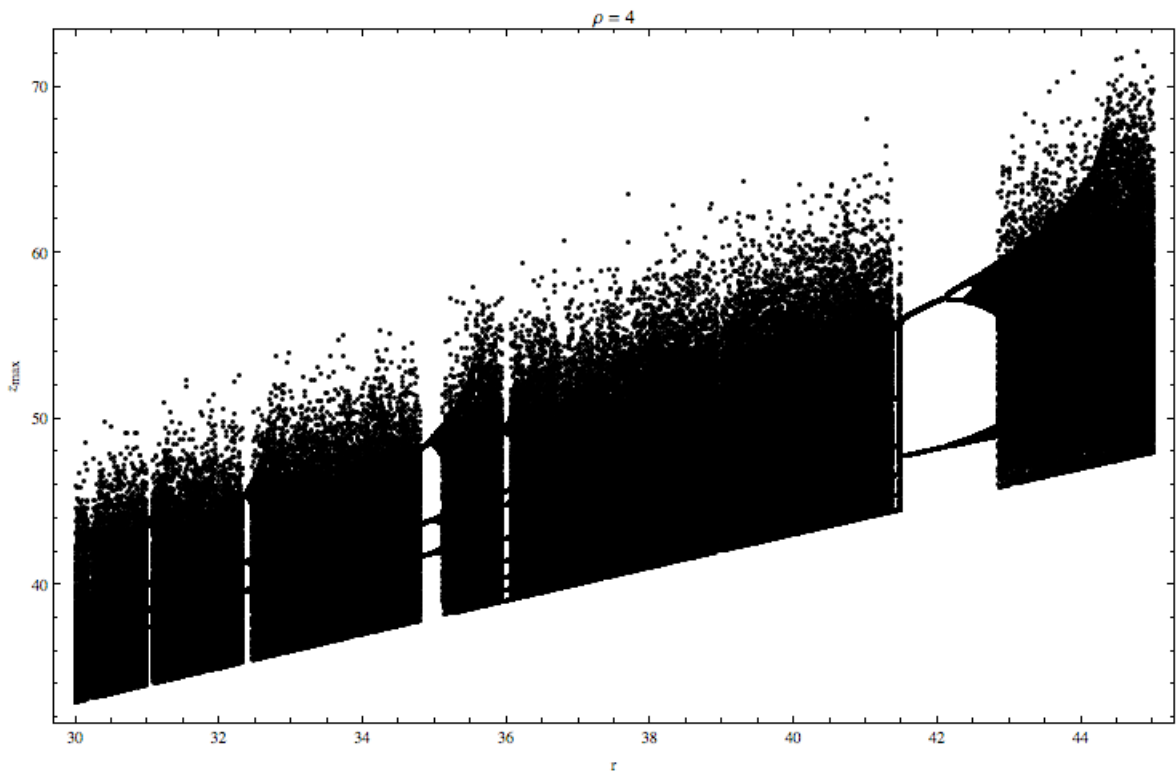
Figure 4.6: Z orbit diagram for $\rho = 4$. There are numerous ranges of r where there is limit cycle behavior.

5

Conclusion

We have seen regions of r for which there are fixed points, limit cycles, and chaotic motion for both the symmetric and asymmetric inflow of water. For the asymmetric case with nonzero ρ, the bifurcation from one to three fixed points was imperfect and occurred at larger values of r for larger ρ values. The asymmetry was clearly visible in the plots of x versus t for $\rho > 0$, which showed the wheel spending more time with positive angular velocity. In both the symmetric and asymmetric case, sensitive dependence on initial conditions was observed. By examining the plots of $\ln[d(t)]$ versus t, we saw that the exponential divergence was not as strong for the asymmetric wheel. Period doubling was present for both types of inflows in regions of r with limit cycle behavior. As the inflow of water gets more asymmetric, the wheel shows this by turning more in one direction than the other. The chaotic behavior is dampened by the asymmetry.

Bibliography

[1] Thornton and Marion, 2004: *Classical Dynamics of Particles and Systems.* Brooks/Cole.

[2] Tél, Tamás and Gruiz, Márton, 2006: *Chaotic Dynamics.* Cambridge University Press.

[3] Kundu and Cohen, 2008: *Fluid Mechanics.* Elsevier, Inc.

[4] Strogatz, S. H., 1994: *Nonlinear Dynamics and Chaos.* Perseus Books Publishing, LLC.

[5] Lorenz, Edward N., 1963: Deterministic Nonperiodic Flow. *Journal of the Atmospheric Sciences*, **20**, 130-141.

[6] Wolfram Mathematica 8 For Students. http://www.wolfram.com/mathematica/

[7] http://www.soe.ucsc.edu/classes/ams214/Winter10/index.html

[8] http://physics.ucsc.edu/~peter/115/

6

Appendix: Mathematica Code

This appendix gives the code used in Wolfram Mathematica 8 For Students to obtain solutions of the water wheel equations and create all of the plots.

Figure 2.1:

```
g1 = Show[{Plot[{Sqrt[r - 1], -Sqrt[r - 1]}, {r, 1, 10},
     PlotStyle -> Black]}, {Plot[0, {r, 0, 1},
     PlotStyle -> Black]}, {Plot[0, {r, 1, 10},
     PlotStyle -> {Black, Dashed}]}, Axes -> False, Frame -> True,
   PlotRange -> {{0, 10}, {-4, 4}}, FrameLabel -> {"r", "x"},
   PlotLabel -> "\[Rho] = 0"];
g2 = ContourPlot[(x^2) - (1/4)/x + 1 == r, {r, 0, 10}, {x, -4, 4},
   Frame -> False, Axes -> True, AxesLabel -> {"r", "x"},
   WorkingPrecision -> 1000000, PlotLabel -> "\[Rho] = 0.25"];
g3 = ContourPlot[(x^2) - 1/x + 1 == r, {r, 0, 10}, {x, -4, 4},
   Frame -> False, Axes -> True, AxesLabel -> {"r", "x"},
   WorkingPrecision -> 1000000, PlotLabel -> "\[Rho] = 1"];
g4 = ContourPlot[(x^2) - 4/x + 1 == r, {r, 0, 10}, {x, -4, 4},
   Frame -> False, Axes -> True, AxesLabel -> {"r", "x"},
   WorkingPrecision -> 1000000, PlotLabel -> "\[Rho] = 4"];
Show[GraphicsGrid[{{g1, g2}, {g3, g4}}]]
```

Figure 2.2:

```
Show[{Plot[x^3 - x (8 - 1), {x, -3.5, 3.5}],
  Plot[{-2, 2, 4, 6, 8, (14/3) Sqrt[7/3]}, {x, -3.5, 3.5},
   PlotStyle -> Dashed]}, PlotRange -> {{-3.5, 3.5}, {-9, 9}},
```

```
AxesLabel -> {"x", " "}]
```

Figure 2.3:

```
Plot[{(2/3) (r - 1) Sqrt[(r - 1)/3], -(2/3) (r - 1) Sqrt[(r - 1)/
     3]}, {r, 0, 10}, AxesLabel -> {"r", "\[Rho]"},
 PlotStyle -> Black]
```

Figure 2.4:

```
ContourPlot3D[
 x^3 - x (r - 1) - \[Rho] == 0, {x, -7, 7}, {r, -10,
  10}, {\[Rho], -12, 12}, AxesLabel -> {"x", "r", "\[Rho]"}]
```

Figure 3.1:

```
g1 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
      y'[t] == (.75) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 1, y[0] == 1,
      z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> 1000000]], {t,
     0, 50}, AxesLabel -> {"t", "x(t)"},
   PlotRange -> {{0, 20}, {-0.5, 1.5}}, PlotLabel -> "r = 0.75"];
g2 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
      y'[t] == (13) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 1, y[0] == 1,
      z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> 1000000]], {t,
     0, 60}, AxesLabel -> {"t", "x(t)"},
   PlotRange -> {{0, 60}, {-10, 10}}, PlotLabel -> "r = 13"];
g3 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
      y'[t] == (27.65) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 1, y[0] == 1,
      z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> 1000000]], {t,
     0, 90}, AxesLabel -> {"t", "x(t)"},
   PlotRange -> {{0, 65}, {-20, 20}}, PlotLabel -> "r = 27.65"];
g4 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
      y'[t] == (29) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 1, y[0] == 1,
```

```
        z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> 1000000]], {t,
      0, 90}, AxesLabel -> {"t", "x(t)"},
     PlotRange -> {{0, 70}, {-20, 20}}, PlotLabel -> "r = 29"];
Show[GraphicsGrid[{{g1, g2}, {g3, g4}}]]
```

Figure 3.2

```
h1 = ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (.75) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == -3, y[0] == 4,
       z[0] == 20}, {x, y, z}, {t, 0, 150}, MaxSteps -> 1000000]], {t,
      0, 30}, AxesLabel -> {"x", "y", "z"},
     PlotRange -> {{-9, 10}, {-7, 6}, {-1, 22}},
     PlotLabel -> "r = 0.75"];
h2 = ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (13) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == -3, y[0] == 4,
       z[0] == 20}, {x, y, z}, {t, 0, 150}, MaxSteps -> 1000000]], {t,
      0, 35}, AxesLabel -> {"x", "y", "z"},
     PlotRange -> {{-5, 10}, {-2, 7}, {5, 22}}, PlotLabel -> "r = 13"];
h3 = ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (27.65) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == -3, y[0] == 4,
       z[0] == 20}, {x, y, z}, {t, 0, 300}, MaxSteps -> 1000000]], {t,
      250, 300}, AxesLabel -> {"x", "y", "z"},
     PlotRange -> {{-20, 20}, {-20, 20}, {10, 40}},
     PlotLabel -> "r = 27.65"];
h4 = ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == -3, y[0] == 4,
       z[0] == 20}, {x, y, z}, {t, 0, 350}, MaxSteps -> 1000000]], {t,
      250, 300}, AxesLabel -> {"x", "y", "z"},
     PlotRange -> {{-20, 20}, {-20, 20}, {9, 50}},
     PlotLabel -> "r = 29"];
Show[GraphicsGrid[{{h1, h2}, {h3, h4}}]]
```

Figure 3.3:

```
Show[{ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == Sqrt[28],
       y[0] == Sqrt[28], z[0] == 28}, {x, y, z}, {t, 0, 100},
      MaxSteps -> 1000000]], {t, 0, 50},
   PlotStyle -> {Thickness[.015], Red}],
  ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 0, y[0] == 1,
       z[0] == 49}, {x, y, z}, {t, 0, 100}, MaxSteps -> 1000000]], {t,
     0, 55}],
  ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == -Sqrt[28],
       y[0] == -Sqrt[28], z[0] == 28}, {x, y, z}, {t, 0, 100},
      MaxSteps -> 1000000]], {t, 0, 50},
   PlotStyle -> {Thickness[.015], Red}],
  ParametricPlot3D[
   Evaluate[{x[t], y[t], z[t]} /.
     NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 0, y[0] == 0,
       z[0] == 0}, {x, y, z}, {t, 0, 100}, MaxSteps -> 1000000]], {t,
    0, 50}, PlotStyle -> {Thickness[.015], Black}]},
 PlotRange -> {{-20, 20}, {-20, 20}, {-1, 50}}, PlotLabel -> "r = 29",
  AxesLabel -> {"x", "y", "z"}]
```

Figure 3.4:

```
j1 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 1, y[0] == 1,
       z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> 1000000]], {t,
     0, 60}, AxesLabel -> {"t", "x(t)"}, PlotStyle -> Blue];
j2 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
```

```
        z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 1.00000001,
          y[0] == 1, z[0] == 1}, {x, y, z}, {t, 0, 1000},
        MaxSteps -> 1000000]], {t, 0, 60}, AxesLabel -> {"t", "x(t)"},
    PlotStyle -> Black];
Show[j1, j2]
```

Figure 3.5:

```
sol = NDSolve[{x'[t] == 10 (y[t] - x[t]),
    y'[t] == (29) x[t] - x[t] z[t] - y[t], z'[t] == x[t] y[t] - z[t],
    x[0] == 1, y[0] == 1, z[0] == 1}, {x, y, z}, {t, 0, 1000},
  MaxSteps -> Infinity]; xx[t_] := x[t] /. sol[[1]];
yy[t_] := y[t] /. sol[[1]]; zz[t_] := z[t] /. sol[[1]];
sol1 = NDSolve[{x'[t] == 10 (y[t] - x[t]),
    y'[t] == (29) x[t] - x[t] z[t] - y[t], z'[t] == x[t] y[t] - z[t],
    x[0] == xx[1000], y[0] == yy[1000], z[0] == zz[1000]}, {x, y,
    z}, {t, 0, 1000}, MaxSteps -> Infinity];
x1[t_] := x[t] /. sol1[[1]]; y1[t_] := y[t] /. sol1[[1]];
z1[t_] := z[t] /. sol1[[1]];
sol2 = NDSolve[{x'[t] == 10 (y[t] - x[t]),
    y'[t] == (29) x[t] - x[t] z[t] - y[t], z'[t] == x[t] y[t] - z[t],
    x[0] == xx[1000] + .00000001, y[0] == yy[1000],
    z[0] == zz[1000]}, {x, y, z}, {t, 0, 1000}, MaxSteps -> Infinity];
x2[t_] := x[t] /. sol2[[1]]; y2[t_] := y[t] /. sol2[[1]];
z2[t_] := z[t] /. sol2[[1]];
ListPlot[Table[{i,
    Log[Sqrt[(x2[i] - x1[i])^2 + (y2[i] - y1[i])^2 + (z2[i] -
        z1[i])^2]]}, {i, 0, 55, .001}], Axes -> False, Frame -> True,
  FrameLabel -> {"t", "ln[d(t)]"}, PlotStyle -> Black]
```

Figure 3.6:

```
ptslists1 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t]),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 5, y[0] == 8,
      z[0] == 3}, {x, y, z}, {t, 0, 300}, MaxSteps -> Infinity];
    f[t_] := x[t] /. sol[[1]];
    pts = Table[{r, f[i]}, {i, 250, 300, .1}], {r, 0, 35, 0.1}];
ptslists2 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t]),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == -3, y[0] == -1,
      z[0] == -1}, {x, y, z}, {t, 0, 300}, MaxSteps -> Infinity];
```

```
    f[t_] := x[t] /. sol[[1]];
    pts = Table[{r, f[i]}, {i, 250, 300, .1}], {r, 0, 35, 0.1}];
ptslists3 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t]),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (0) x[t], x[0] == 1, y[0] == 1,
      z[0] == 1}, {x, y, z}, {t, 0, 300}, MaxSteps -> Infinity];
    f[t_] := x[t] /. sol[[1]];
    pts = Table[{r, f[i]}, {i, 250, 300, .1}], {r, 0, 35, 0.1}];
Show[{ListPlot[ptslists1, PlotStyle -> Black],
  ListPlot[ptslists2, PlotStyle -> Black],
  ListPlot[ptslists3, PlotStyle -> Black],
  Plot[{Sqrt[r - 1], -Sqrt[r - 1]}, {r, 1, 15}, PlotStyle -> Red],
  Plot[0, {r, 0, 1}, PlotStyle -> Red],
  Plot[0, {r, 1, 15}, PlotStyle -> {Purple, Dashed}]}, Axes -> False,
 Frame -> True, FrameLabel -> {"r", "x"}]
```

Figure 3.7:

```
Plot[Evaluate[
  z[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
    y'[t] == (29) x[t] - x[t] z[t] - y[t],
    z'[t] == x[t] y[t] - z[t] + 0 x[t], x[0] == 2, y[0] == 0,
    z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> Infinity]], {t,
  0, 55}, AxesLabel -> {"t", "z(t)"}]
```

Figure 3.8:

```
sol = NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
   y'[t] == 29 x[t] - x[t] z[t] - y[t],
   z'[t] == x[t] y[t] - z[t] + 0 x[t], x[0] == 1, y[0] == 1,
   z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> Infinity];
f[t_] := z[t] /. sol[[1]]; n = 1; zvals =
 Table[f[i], {i, 500, 1000, .01}]; zmax =
 Table[0, {i, 0, Length[zvals] - 1}]; For[j = 2, j < Length[zvals],
 j = j + 1,
 If[(zvals[[j]] > zvals[[j - 1]]) && (zvals[[j]] > zvals[[j + 1]]),
  zmax[[n]] = zvals[[j]]; n = n + 1]]; pts =
 Table[{zmax[[l]], zmax[[l + 1]]}, {l, 1, n - 2}]; Show[
 ListPlot[pts], {Axes -> False, Frame -> True,
  FrameLabel -> {"\!\(\*SubscriptBox[\(z\), \(n\)]\)",
    "\!\(\*SubscriptBox[\(z\), \(n + 1\)]\)"}}]
```

Figure 3.9:

```
sol = NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
   y'[t] == 26.95 x[t] - x[t] z[t] - y[t],
   z'[t] == x[t] y[t] - z[t] + 0 x[t], x[0] == 1, y[0] == 1,
   z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> Infinity];
f[t_] := z[t] /. sol[[1]]; n = 1; zvals =
 Table[f[i], {i, 500, 1000, .01}]; zmax =
 Table[0, {i, 0, Length[zvals] - 1}]; For[j = 2, j < Length[zvals],
 j = j + 1,
 If[(zvals[[j]] > zvals[[j - 1]]) && (zvals[[j]] > zvals[[j + 1]]),
  zmax[[n]] = zvals[[j]]; n = n + 1]]; pts =
 Table[{zmax[[l]], zmax[[l + 1]]}, {l, 1, n - 2}]; q1 =
 Show[ListPlot[pts], {Axes -> False, Frame -> True,
   FrameLabel -> {"\!\(\*SubscriptBox[\(z\), \(n\)]\)",
     "\!\(\*SubscriptBox[\(z\), \(n + 1\)]\)"},
   PlotLabel -> "r = 26.95"}];
sol = NDSolve[{x'[t] == 10 (y[t] - x[t] + 0),
   y'[t] == 27.65 x[t] - x[t] z[t] - y[t],
   z'[t] == x[t] y[t] - z[t] + 0 x[t], x[0] == 1, y[0] == 1,
   z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> Infinity];
f[t_] := z[t] /. sol[[1]]; n = 1; zvals =
 Table[f[i], {i, 500, 1000, .01}]; zmax =
 Table[0, {i, 0, Length[zvals] - 1}]; For[j = 2, j < Length[zvals],
 j = j + 1,
 If[(zvals[[j]] > zvals[[j - 1]]) && (zvals[[j]] > zvals[[j + 1]]),
  zmax[[n]] = zvals[[j]]; n = n + 1]]; pts =
 Table[{zmax[[l]], zmax[[l + 1]]}, {l, 1, n - 2}]; q2 =
 Show[ListPlot[pts], {Axes -> False, Frame -> True,
   FrameLabel -> {"\!\(\*SubscriptBox[\(z\), \(n\)]\)",
     "\!\(\*SubscriptBox[\(z\), \(n + 1\)]\)"},
   PlotLabel -> "r = 27.65"}];
GraphicsRow[{q1, q2}]
```

Figures 3.10 and 3.11:

```
ptslists =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t]),
      y'[t] == (r) x[t] - x[t] z[t] - y[t], z'[t] == x[t] y[t] - z[t],
       x[0] == 1, y[0] == 1, z[0] == 1}, {x, y, z}, {t, 0, 1000},
     MaxSteps -> Infinity]; f[t_] := z[t] /. sol[[1]]; n = 1;
   zvals = Table[f[i], {i, 700, 1000, .01}];
   zmax = Table[0, {i, 0, Length[zvals] - 1}];
   For[j = 2, j < Length[zvals], j = j + 1,
    If[(zvals[[j]] > zvals[[j - 1]]) && (zvals[[j]] > zvals[[j + 1]]),
      zmax[[n]] = zvals[[j]]; n = n + 1]];
   pts = Table[{r, zmax[[l]]}, {l, 1, n - 2}], {r, 26.3, 28, 0.0025}];
```

```
ListPlot[ptslists, {Axes -> False, Frame -> True, PlotStyle -> Black,
  FrameLabel -> {"r", "\!\(\*SubscriptBox[\(z\), \(max\)]\)"},
  PlotRange -> {{26.3, 28}, {30, 41}}}]
ListPlot[ptslists, {Axes -> False, Frame -> True, PlotStyle -> Black,
  FrameLabel -> {"r", "\!\(\*SubscriptBox[\(z\), \(max\)]\)"},
  PlotRange -> {{26.7, 27.3}, {37.5, 38.8}}}]
```

Figure 3.12:

```
Clear["Global`*"]
ptslists =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t]),
      y'[t] == (r) x[t] - x[t] z[t] - y[t], z'[t] == x[t] y[t] - z[t],
        x[0] == 1, y[0] == 1, z[0] == 1}, {x, y, z}, {t, 0, 1000},
      MaxSteps -> Infinity]; f[t_] := z[t] /. sol[[1]]; n = 1;
   zvals = Table[f[i], {i, 700, 1000, .01}];
   zmax = Table[0, {i, 0, Length[zvals] - 1}];
   For[j = 2, j < Length[zvals], j = j + 1,
    If[(zvals[[j]] > zvals[[j - 1]]) && (zvals[[j]] > zvals[[j + 1]]),
      zmax[[n]] = zvals[[j]]; n = n + 1]];
   pts = Table[{r, zmax[[l]]}, {l, 1, n - 2}], {r, 25, 45, 0.025}];
ListPlot[ptslists, {Axes -> False, Frame -> True, PlotStyle -> Black,
  FrameLabel -> {"r", "\!\(\*SubscriptBox[\(z\), \(max\)]\)"}}]
```

Figure 4.1:

```
s1 = Plot[
  Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + .7),
      y'[t] == (29) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (.7) x[t], x[0] == 1, y[0] == 1,
      z[0] == 1}, {x, y, z}, {t, 0, 200}, MaxSteps -> 1000000]], {t,
   0, 100}, AxesLabel -> {"t", "x(t)"},
  PlotRange -> {{0, 50}, {-18, 20}},
  PlotLabel -> "(a)   \[Rho] = .7"];
s2 = Plot[
  Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 2),
      y'[t] == (29) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (2) x[t], x[0] == 1, y[0] == 1,
      z[0] == 1}, {x, y, z}, {t, 0, 200}, MaxSteps -> 1000000]], {t,
   0, 100}, AxesLabel -> {"t", "x(t)"},
  PlotRange -> {{0, 50}, {-15, 21}},
  PlotLabel -> "(b)   \[Rho] = 2"];
s3 = Plot[
```

```
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (4) x[t], x[0] == 1, y[0] == 1,
       z[0] == 1}, {x, y, z}, {t, 0, 200}, MaxSteps -> 1000000]], {t,
    0, 100}, AxesLabel -> {"t", "x(t)"},
   PlotRange -> {{0, 50}, {-15, 21}},
   PlotLabel -> "(c)   \[Rho] = 4"];
s4 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 7),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (7) x[t], x[0] == 1, y[0] == 1,
       z[0] == 1}, {x, y, z}, {t, 0, 200}, MaxSteps -> 1000000]], {t,
    0, 100}, AxesLabel -> {"t", "x(t)"},
   PlotRange -> {{0, 75}, {-10, 22}},
   PlotLabel -> "(d)   \[Rho] = 7"];
Show[GraphicsGrid[{{s1, s2}, {s3, s4}}]]
```

Figure 4.2:

```
Clear["Global`*"]
j1 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (4) x[t], x[0] == 1, y[0] == 1,
       z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> 1000000]], {t,
     30, 110}, PlotStyle -> Blue];
j2 = Plot[
   Evaluate[
    x[t] /. NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
       y'[t] == (29) x[t] - x[t] z[t] - y[t],
       z'[t] == x[t] y[t] - z[t] + (4) x[t], x[0] == 1.00000001,
       y[0] == 1, z[0] == 1}, {x, y, z}, {t, 0, 1000},
      MaxSteps -> 1000000]], {t, 30, 110}, PlotStyle -> Black];
Show[{j1, j2}, Axes -> False, Frame -> True,
 FrameLabel -> {"t", "x(t)"}, PlotRange -> {{30, 110}, {-15, 15}},
 PlotLabel -> "\[Rho] = 4"]
```

Figure 4.3:

```
Clear["Global`*"]
sol = NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
   y'[t] == (29) x[t] - x[t] z[t] - y[t],
   z'[t] == x[t] y[t] - z[t] + 4 x[t], x[0] == 1, y[0] == 1,
```

```
        z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> Infinity];
xx[t_] := x[t] /. sol[[1]]; yy[t_] := y[t] /. sol[[1]];
zz[t_] := z[t] /. sol[[1]];
sol1 = NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
    y'[t] == (29) x[t] - x[t] z[t] - y[t],
    z'[t] == x[t] y[t] - z[t] + 4 x[t], x[0] == xx[1000],
    y[0] == yy[1000], z[0] == zz[1000]}, {x, y, z}, {t, 0, 1000},
  MaxSteps -> Infinity]; x1[t_] := x[t] /. sol1[[1]];
y1[t_] := y[t] /. sol1[[1]]; z1[t_] := z[t] /. sol1[[1]];
sol2 = NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
    y'[t] == (29) x[t] - x[t] z[t] - y[t],
    z'[t] == x[t] y[t] - z[t] + 4 x[t], x[0] == xx[1000] + .00000001,
    y[0] == yy[1000], z[0] == zz[1000]}, {x, y, z}, {t, 0, 1000},
  MaxSteps -> Infinity]; x2[t_] := x[t] /. sol2[[1]];
y2[t_] := y[t] /. sol2[[1]]; z2[t_] := z[t] /. sol2[[1]];
ListPlot[Table[{i,
   Log[Sqrt[(x2[i] - x1[i])^2 + (y2[i] - y1[i])^2 + (z2[i] -
        z1[i])^2]]}, {i, 0, 110, .005}], Axes -> False,
 Frame -> True, FrameLabel -> {"t", "ln[d(t)]"}, PlotStyle -> Black,
 PlotLabel -> "\[Rho] = 4"]
```

Figure 4.4:

```
Clear["Global`*"]
ptslists1 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t] + .1),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (.1) x[t], x[0] == 0, y[0] == 0,
      z[0] == 0}, {x, y, z}, {t, 0, 500}, MaxSteps -> Infinity];
   f[t_] := x[t] /. sol[[1]];
   pts = Table[{r, f[i]}, {i, 450, 500, .1}], {r, 0, 35, 0.1}];
ptslists2 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t] + .1),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (.1) x[t], x[0] == -3, y[0] == -1,
      z[0] == -11}, {x, y, z}, {t, 0, 500}, MaxSteps -> Infinity];
   f[t_] := x[t] /. sol[[1]];
   pts = Table[{r, f[i]}, {i, 450, 500, .1}], {r, 0, 35, 0.1}];
ptslists3 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t] + .1),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (.1) x[t], x[0] == 4, y[0] == -2,
      z[0] == 10}, {x, y, z}, {t, 0, 500}, MaxSteps -> Infinity];
```

```
    f[t_] := x[t] /. sol[[1]];
    pts = Table[{r, f[i]}, {i, 450, 500, .1}], {r, 0, 35, 0.1}];
Show[{ListPlot[ptslists1, PlotStyle -> Black],
  ListPlot[ptslists2, PlotStyle -> Black],
  ListPlot[ptslists3, PlotStyle -> Black],
  ContourPlot[
   x^3 - x (r - 1) - .1 == 0, {r, 0, 15.5}, {x, -4, 8}]}, {Axes ->
   False, Frame -> True, FrameLabel -> {"r", "x"},
  PlotRange -> {{-1, 36}, {-20, 20}}, PlotLabel -> "\[Rho] = 0.1"}]
```

Figure 4.5:

```
Clear["Global`*"]
ptslists1 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (4) x[t], x[0] == 5, y[0] == 8,
      z[0] == 3}, {x, y, z}, {t, 0, 500}, MaxSteps -> Infinity];
   f[t_] := x[t] /. sol[[1]];
   pts = Table[{r, f[i]}, {i, 450, 500, .1}], {r, 0, 40, 0.1}];
ptslists2 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (4) x[t], x[0] == -1, y[0] == -1,
      z[0] == -1}, {x, y, z}, {t, 0, 500}, MaxSteps -> Infinity];
   f[t_] := x[t] /. sol[[1]];
   pts = Table[{r, f[i]}, {i, 450, 500, .1}], {r, 0, 40, 0.1}];
ptslists3 =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + (4) x[t], x[0] == -11, y[0] == 11,
      z[0] == -10}, {x, y, z}, {t, 0, 500}, MaxSteps -> Infinity];
   f[t_] := x[t] /. sol[[1]];
   pts = Table[{r, f[i]}, {i, 450, 500, .1}], {r, 0, 40, 0.1}];
Show[{ListPlot[ptslists1, PlotStyle -> Black],
  ListPlot[ptslists2, PlotStyle -> Black],
  ListPlot[ptslists3, PlotStyle -> Black],
  ContourPlot[
   x^3 - x (r - 1) - 4 == 0, {r, 0, 27.3}, {x, -4, 8}]}, {Axes ->
   False, Frame -> True, FrameLabel -> {"r", "x"},
  PlotRange -> {{-1, 41}, {-25, 25}}, PlotLabel -> "\[Rho] = 4"}]
```

Figure 4.6:

```
Clear["Global`*"]
ptslists =
  Table[sol =
    NDSolve[{x'[t] == 10 (y[t] - x[t] + 4),
      y'[t] == (r) x[t] - x[t] z[t] - y[t],
      z'[t] == x[t] y[t] - z[t] + 4 x[t], x[0] == 1, y[0] == 1,
      z[0] == 1}, {x, y, z}, {t, 0, 1000}, MaxSteps -> Infinity];
   f[t_] := z[t] /. sol[[1]]; n = 1;
   zvals = Table[f[i], {i, 700, 1000, .01}];
   zmax = Table[0, {i, 0, Length[zvals] - 1}];
   For[j = 2, j < Length[zvals], j = j + 1,
    If[(zvals[[j]] > zvals[[j - 1]]) && (zvals[[j]] > zvals[[j + 1]]),
      zmax[[n]] = zvals[[j]]; n = n + 1]];
   pts = Table[{r, zmax[[l]]}, {l, 1, n - 2}], {r, 30, 45, 0.015}];
ListPlot[ptslists, {Axes -> False, Frame -> True, PlotStyle -> Black,
  FrameLabel -> {"r", "\!\(\*SubscriptBox[\(z\), \(max\)]\)"},
  PlotLabel -> "\[Rho] = 4"}]
```