Chaotic Pendulum

Experiment CP

University of Florida — Department of Physics
PHY4803L — Advanced Physics Laboratory

Objective

Both regular and chaotic motion for a spring-pendulum system will be measured and analyzed. You will learn in your investigations how to recognize systems for which chaotic behavior can be expected, how to display data arising from these systems, how to quantify features of the dynamical variables involved, and how to perform computer simulations and comparisons with experimental data.

References

Henry D. I. Abarbanel, *The analysis of observed chaotic data in physical systems,* Reviews of Modern Physics, **65**, 1331-1392 (1993).

Gregory L. Baker, Jerry P. Gollub, *Chaotic Dynamics: An Introduction,* 2nd ed., Cambridge University Press, New York, 1996.

Francis C. Moon, *Chaotic and Fractal Dynamics,* John Wiley and Sons, New York, 1992.

Introduction

The harmonic oscillator is a paradigm of predictability. The resonance response of a harmonic oscillator can be exploited to make clocks accurate to 1 part in 10^{14}. Harmonic oscillations are normally observed in systems operating near equilibrium and governed by linear or "Hooke's law" restoring forces. Nonetheless, for small enough excursions from equilibrium, systems governed by nonlinear restoring forces can also display harmonic oscillations. A good example is the pendulum. When its motion stays near equilibrium (at $\theta = 0$), the nonlinear restoring torque (proportional to $\sin\theta$) behaves like a linear restoring torque (proportional to θ) and the pendulum executes simple harmonic motion. However, when driven strongly, the excursions from equilibrium can grow into regions where the nonlinearity becomes important. The predictable pattern of repeating oscillations might give way to chaos—non-repeating motion characterized by a particular kind of unpredictability.

Chaotic systems are completely deterministic. For example, our apparatus is governed by equations easily derived from $F = ma$. The unpredictability arises not from any inherent randomness but rather from an extreme sensitivity to initial conditions. Harmonic oscillators ultimately settle on some well defined state of motion. Momentarily perturbed from this "steady state" motion, they will return to it according to a predictable "transient" time constant. In contrast, perturb a nonlinear oscillator and the difference between the

perturbed and unperturbed states of motion can grow exponentially.

You will investigate just such a nonlinear oscillator. Illustrated in Fig. 1, this spring-pendulum apparatus can be configured to display both chaotic and non-chaotic behavior. The non-chaotic data sets will be graphed and analyzed using traditional techniques. The chaotic data sets will be graphed in Poincaré sections and quantified by fractal dimensions and Lyapunov exponents.

Exercise 1 *The applicability of the superposition principle is the distinguishing property between linear and nonlinear systems. Show that solutions to the pendulum equation,* $d^2\theta/dt^2 = -(g/l)\sin\theta$*, obey the superposition principle only if one can assume* $\sin\theta = \theta$ *throughout the motion. That is, if* $\theta_1(t)$ *and* $\theta_2(t)$ *are solutions, show that* $\theta_1 + \theta_2$ *is a solution when* $\sin\theta = \theta$*, but not otherwise.*

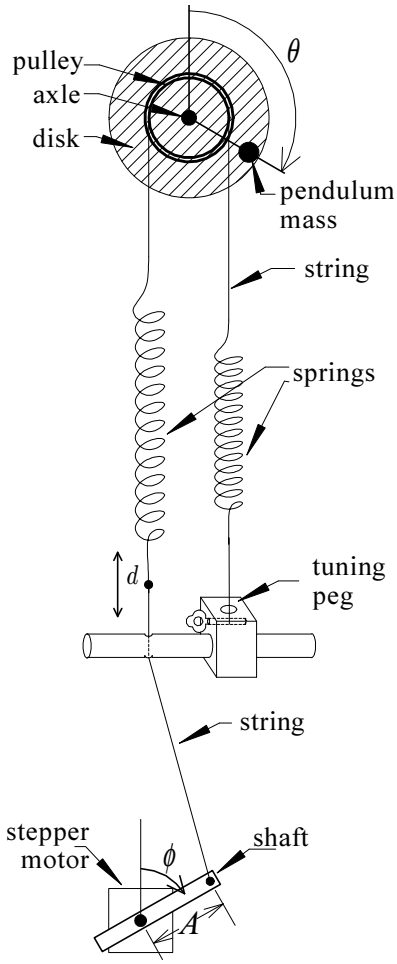

Figure 1: The chaotic pendulum apparatus. Not shown: electronics, supports, and the rotary encoder housing the axle.

Apparatus

The apparatus consists of a physical pendulum coupled to a drive motor using a pair of springs, a pulley, and string. The physical pendulum consists of an aluminum disk, a removable pendulum mass, a pulley, and an axle. The axle and pulley are those of the Pasco Rotary Motion Sensor which has a rotary encoder for measuring the pendulum angle. A computer equipped with a National Instruments PCI 6601 timer/counter board controls the drive motor and reads the rotary encoder. The computer is also used to save, display, and analyze the resulting data.

The drive system consists of a stepper motor, shaft, and controller. A uniform pulse train is sent by the computer to the controller which sequentially energizes the various motor windings to effect a single angular step of the motor shaft for each pulse. The stepper motor requires 200 pulses to complete one revolution. As the stepper motor rotates, the driven end of the spring oscillates, keeping the pendulum in motion.

For measuring the pendulum angle, the rotary encoder emits logic pulses (1440/rev) on two separate phonojacks for clockwise and counterclockwise rotations. The computer is capable of direct up-down counting with these signals. The count, directly proportional to the rotation angle, is read with every stepper motor pulse and saved into an array. Because

the same pulses initiate a reading of the angular count and step the motor, the pendulum angle and the drive angle are simultaneously determined.

The raw counter readings, spaced 1/200th of the drive period apart in time, are scaled and numerically smoothed and differentiated using Savitsky-Golay filtering.[1] The filters are equivalent to a separate least squares fit to a polynomial for each data point with the fitting region symmetrically surrounding the point. They are used to determine the angle θ, the angular velocity $\omega = d\theta/dt$, and the angular acceleration $\alpha = d^2\theta/dt^2$.

Equation of Motion

As shown in Fig. 1, the pendulum angle θ is defined from the vertical with $\theta = 0$ for the straight-up, or inverted, position. Clockwise angles are taken positive.

Also illustrated on Fig. 1, the drive phase ϕ is taken as the angle of the motor shaft from vertical. The uniform pulse train sent to the motor has an adjustable rate and will cause the shaft to rotate at a constant angular velocity Ω so that

$$\phi = \Omega t \tag{1}$$

where the shaft is assumed to be straight up at $t = 0$. Then, the displacement d of the driven end of the spring can be taken to be[2]

$$d = A \cos \Omega t \tag{2}$$

[1] Abraham Savitsky and Marcel J. E. Golay, *Smoothing and differentiation of data by simplified least squares procedures,* Analytical Chemistry, **36** pp 1627-39, 1964.

[2] With the distance from the center of the drive motor to the guide hole given by L, the driven end of the spring moves according to $d = A\cos\phi - \frac{A^2}{4L}(1-\cos 2\phi)$. The offset term, $-A^2/4L$, is typically no larger than 2 mm and the term proportional to $\cos 2\phi$ causes a small amplitude driving oscillation at an angular frequency of 2Ω which will be ignored in the analysis.

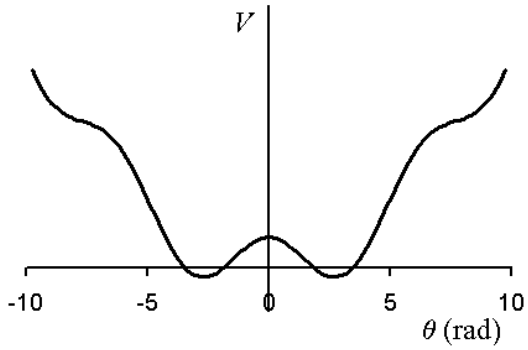

Figure 2: Near scale drawing of the potential $V(\theta)$ for our system.

The disk, pulley and axle have a moment of inertia I_0. They are well balanced and cause virtually no torque in a gravitational field. The pendulum mass m placed a distance l from the rotation axis produces a gravitational torque $mgl \sin\theta$. It also increases the moment of inertia to $I = I_0 + I_m + ml^2$, where I_m is its principal moment of inertia.

The string between the two springs rides over the pulley without slipping, and the relatively weak springs stay elongated at all times. The springs/strings/pulley arrangement is adjusted as described later so that the springs have equal stretch (provide equal tension) when $\theta = 0$ and $d = 0$. In this inverted configuration, the pendulum would have no torque acting on it, but it would be in an unstable equilibrium, ready to fall to the left or right.

Exercise 2 *(a) Show that the torque arising from the two springs and from gravity acting on the pendulum mass is given by*

$$\tau_c = -2kr^2\theta + krd + mgl \sin\theta \tag{3}$$

where k is the force constant for each spring and r is the pulley radius. (b) Show that for $d = 0$, (i.e., $A = 0$, which means the pendulum is undriven) the potential energy is given by

$$V = kr^2\theta^2 + mgl \cos\theta \tag{4}$$

(c) Show that for $mgl \leq 2kr^2$, $V(\theta)$ has one minimum at $\theta = 0$; and that for $mgl > 2kr^2$, $V(\theta)$ has a maximum at $\theta = 0$ and two equally deep minima at $\pm\theta_e$ where

$$\frac{\sin\theta_e}{\theta_e} = \frac{2kr^2}{mgl} \tag{5}$$

As shown in Fig. 2, our system potential has a double well shape.

Dissipation

The torque τ_c (Eq. 3) arises from the conservative forces of gravity and Hooke's law. There are also nonconservative torques due to eddy current damping and axle friction through which energy is continually drained or dissipated from the system. If this energy is not continually replenished, for example, by the stepper motor drive system, the motion slows and ultimately stops. A damping torque proportional to and opposite the angular velocity is created by induced eddy-currents when a neodymium magnet is brought near the rotating aluminum disk. This torque can be modeled as $-b\,\omega$. Friction, arising largely in the axle bearings, causes an additional damping torque of constant magnitude b' that is always opposite the angular velocity. This "axle friction" torque can be modeled $-b'\,\mathrm{sgn}\,\omega$, where $\mathrm{sgn}\,\omega = \omega/|\omega|$. Thus, the net nonconservative torque is given by

$$\tau_f = -b\,\omega - b'\,\mathrm{sgn}\,\omega \tag{6}$$

This form for the damping torque will be verified by fits of data taken from the apparatus. It will also be used in simulations. The (probably small) effect of static friction as the pendulum momentarily stops each time ω goes through zero has not been explored. Occasionally, the entire axle friction term will be dropped in this writeup for readability or whenever analytic solutions are desired. (Including axle friction makes the equation of motion nonlinear and the solutions are not expressible in closed form.)

Exercise 3 *Show that the equation of motion for the apparatus can be written as an autonomous[3] set of three first order differential equations.*

$$\begin{aligned} \dot{\theta} &= \omega && (7)\\ \dot{\omega} &= -\Gamma\omega - \Gamma'\mathrm{sgn}\,\omega - \kappa\theta && (8)\\ &\quad + \mu\sin\theta + \epsilon\cos\phi \\ \dot{\phi} &= \Omega && (9) \end{aligned}$$

where a dot over a variable represents its time derivative and

$$\Gamma = b/I \tag{10}$$

$$\Gamma' = b'/I \tag{11}$$

$$\kappa = 2kr^2/I \tag{12}$$

$$\mu = mgl/I \tag{13}$$

$$\epsilon = krA/I \tag{14}$$

Linear Dynamics

Despite being nonlinear, the system can still behave like a harmonic oscillator as long as the total energy remains small compared to the height of the potential barrier at $\theta = 0$.

Exercise 4 *(a) Show that a Taylor expansion[4] of V (Eq. 4) about either minimum $\theta_0 = \pm\theta_e$ gives*

$$\begin{aligned} V &= kr^2\theta_e^2 + mgl\cos\theta_e && (15)\\ &\quad + \frac{1}{2}(2kr^2 - mgl\cos\theta_e)(\theta-\theta_0)^2 \\ &\quad + \textit{higher order terms} \end{aligned}$$

[3] Autonomous means that time does not appear explicitly on the right sides of the equations.

[4] The first three terms are given by $f(x) = f(x_0) + f'(x_0)(x-x_0) + \frac{1}{2}f''(x_0)(x-x_0)^2$, where primes indicate first and second derivatives with respect to x.

Why is the first order term in $(\theta - \theta_0)$ missing? (b) Show that keeping only up to the quadratic term, (and taking $\Gamma' = 0$), the equations of motion can be written

$$\ddot{\theta}' + \Gamma\dot{\theta}' + \Omega_0^2\theta' = \epsilon \cos \Omega t \tag{16}$$

where a double dot over a variable represents a second derivative with respect to time, $\theta' = \theta - \theta_0$ is the angular displacement from equilibrium, and

$$\Omega_0 = \sqrt{\kappa - \mu \cos \theta_e} \tag{17}$$

is the natural resonance frequency. As discussed below, Eq. 16 is the standard driven harmonic oscillator equation.

Although it is required for chaotic motion, the pendulum mass m can be removed, turning the apparatus into an excellent example of a harmonic oscillator even for large amplitude oscillations. The equation of motion with $m = 0$ (and $\Gamma' = 0$) can be expressed

$$\ddot{\theta} + \Gamma\dot{\theta} + \Omega_0^2\theta = \epsilon \cos \Omega t \tag{18}$$

where $\Omega_0^2 = \kappa$, and Γ, κ, and ϵ are given by Eqs. 10, 12 and 14 (with $I = I_0$).

Exercise 5 *(a) Show that for $\epsilon = 0$ (undriven system), Eq. 18 has an underdamped ($\Gamma/2 < \Omega_0$) solution of the form*

$$\theta = Ce^{-\frac{1}{2}\Gamma t} \cos(\Omega' t + \delta) \tag{19}$$

where

$$\Omega' = \sqrt{\Omega_0^2 - \Gamma^2/4} \tag{20}$$

is the oscillation frequency (pulled slightly below the natural resonance frequency Ω_0 for weak damping). Although you are not asked to do so here, C and δ can be determined from the initial conditions on θ and $\dot{\theta}$ at $t = 0$. (b) Show that for a nonzero ϵ (driven system), the solution to Eq. 18 is the sum of Eq. 19 (called the homogeneous or transient solution) and a particular or steady state solution given by

$$\theta = C' \cos(\Omega t + \delta') \tag{21}$$

where

$$C' = \frac{\epsilon}{\sqrt{(\Omega_0^2 - \Omega^2)^2 + \Omega^2\Gamma^2}} \tag{22}$$

and

$$\tan \delta' = \frac{\Omega\Gamma}{\Omega^2 - \Omega_0^2} \tag{23}$$

Non-linear Dynamics

For chaotic motion, there are no analytic solutions, no closed form expressions for θ vs. t like Eqs. 19 and 21. So how can the measurements be analyzed? The following few sections present some of the techniques for representing, analyzing, and comparing chaotic data sets.

Phase space

The Poincaré-Bendixson theorem states that at least three time dependent (dynamical) variables are required for a system to display chaotic behavior. Our system is at this minimum as demonstrated by the three equations of motion, Eqs. 7-9, one for each variable. Consider carefully how θ, ω, and ϕ are the only time-dependent variables in the system, how they are independent from one another, and how they completely describe the state of the system at any particular point in time.

θ — Angle of the pendulum.

ω — How fast and in what direction the pendulum is moving.

ϕ — Angle of the drive shaft.

The triplet of values (θ, ω, ϕ) can be considered a point $\mathbf{u}$ in a three dimensional coordinate system called *phase space*.

$$\mathbf{u} = \begin{pmatrix} \theta \\ \omega \\ \phi \end{pmatrix} \tag{24}$$

As the three variables change continuously in time, the phase point describing the system moves continuously through phase space. The path through phase space is called a trajectory and a plot of a trajectory is called a phase plot.

The trajectory is predicted by Eqs. 7-9 which can be expressed compactly using the vector notation

$$\dot{\mathbf{u}} = \mathbf{G}(\mathbf{u}) \tag{25}$$

where

$$\dot{\mathbf{u}} = \begin{pmatrix} \dot{\theta} \\ \dot{\omega} \\ \dot{\phi} \end{pmatrix} \tag{26}$$

and $\mathbf{G}(\mathbf{u})$ is a 3-vector function of the three dynamic variables.

$$\mathbf{G}(\mathbf{u}) = \begin{pmatrix} G_\theta(\theta, \omega, \phi) \\ G_\omega(\theta, \omega, \phi) \\ G_\phi(\theta, \omega, \phi) \end{pmatrix} \tag{27}$$

For our system

$$\mathbf{G}(\mathbf{u}) = \tag{28}$$

$$\begin{pmatrix} \omega \\ -\Gamma\omega - \Gamma' \mathrm{sgn}\,\omega - \kappa\theta + \mu \sin\theta + \epsilon \cos\phi \\ \Omega \end{pmatrix}$$

For any point $\mathbf{u}$ in phase space, the equation $\dot{\mathbf{u}} = \mathbf{G}(\mathbf{u})$ predicts the value of $\mathbf{u}$ a short time later. For a small enough time step τ, the future phase can be predicted in terms of the present phase via the definition of the derivative, $\dot{\mathbf{u}} = (\mathbf{u}(t+\tau) - \mathbf{u}(t))/\tau$, as

$$\mathbf{u}(t+\tau) = \mathbf{u}(t) + \tau\, \mathbf{G}(\mathbf{u}(t)) \tag{29}$$

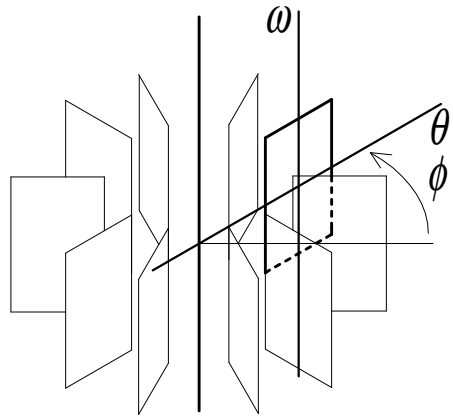

Figure 3: The three-dimensional phase space for the driven nonlinear oscillator. Each rectangle represents a θ-ω plane at a different drive phase ϕ. A darkened plane is shown for one value of ϕ.

To determine $\mathbf{u}$ over a longer time interval, this equation would be iterated using a "small enough" time step τ. With certain refinements to improve accuracy and speed, this is how computer simulations and differential equation solvers work.

The phase space coordinate system can be represented as a continuous set of θ-ω planes encircling a common axis as shown in Fig. 3. Each plane is in one-to-one correspondence with the drive phase ϕ. One turn of the drive system can be thought of as forcing one "orbit" through the planes.

C.Q. 1 *For undriven motion, ϕ is not a dynamical variable for the system and phase space becomes two-dimensional—a single θ-ω plane. Show that the trajectory for undamped, undriven harmonic oscillations lies on an ellipse in the θ-ω phase plane. What relative axes scaling is needed to make the ellipse circular? What determines the direction of motion around the ellipse? What determines the time it takes to go around the ellipse? How do the trajectories depend on the initial conditions?*

Hand sketch at least two different ellipses on the same θ-ω plane.

C.Q. 2 *How does the addition of damping change the trajectories above? How do the trajectories depend on the initial conditions? Hand sketch at least two different trajectories on the same θ-ω plane. Where on the phase plane do all trajectories, no matter the initial conditions, ultimately end up?*

C.Q. 1 demonstrates that oscillations at constant amplitude have a circular trajectory in an appropriately-scaled θ-ω phase plane, while C.Q. 2 demonstrates that the trajectory spirals into the point $(0, 0)$ when damping is present. What do trajectories look like in a driven system, for which phase space is three dimensional—like that of Fig. 3? Figure 4(a) shows a plot of θ vs. t for a driven harmonic oscillator. The fast oscillations at the beginning of the curve correspond to the transient behavior of Eq. 19. These fast oscillations are superimposed on the slower steady state oscillations corresponding to Eq. 21. In plot (b), this motion is shown as a three dimensional phase space trajectory. Starting at the endpoint on the left, the fast transient oscillations become the corkscrew motion about the circular pattern, on which the trajectory repeatedly cycles and representing the steady state oscillations. The steady state orbit is not quite circular, but rides on the surface of a torus as shown in plot (c). With proper scaling, a cross section of the torus would have the two-dimensional circular trajectory of C.Q. 1. But now, while the trajectory cycles around the circle in the θ-ω planes, it simultaneously cycles through the drive phase ϕ—around the long direction of the torus.

Of course, other trajectories are also possible. For example, the natural frequency Ω_0 might be closer to or lower than the drive frequency Ω rather than 20 times higher as in Figs. 4(a) and (b). Or, the damping might be weaker or stronger. These variations require a change in the system parameters. The only variations possible in a given system depend on how the motion is started and determine whether the transient motion starts off large or small. In contrast, the steady state trajectory does not depend on initial conditions—it is an invariant of the system. It is an invariant because the amplitude and phase relations (Eqs. 22 and 23) specify a single closed curve through phase space. Because the phase point repeatedly cycles around a closed curve, the motion is called a *limit cycle*. The motion repeats after a single drive period because the steady state oscillation frequency within the θ-ω planes is the same as the drive frequency through them. Thus, the closed curve is a single loop and called a period-1 limit cycle. Not possible with a linear oscillator, our apparatus can display motion that repeats only after the drive motor has gone around five (or more) times. This would be called a period-5 (or higher) limit cycle.

Attractors

The single invariant curve representing the steady state motion of a driven harmonic oscillator is called an *attractor* for that (three dimensional) system. The final ending point $(0, 0)$ for the damped undriven motion of C.Q. 2 is an attractor for that (two dimensional) system. Attractors are the ultimate fate of the trajectories after transients have decayed away.

Nonlinear oscillators also have attractors. For our system, the type and number of attractors that occur depend on the drive frequency Ω, its amplitude ϵ, and the other system parameters κ, μ, Γ, and Γ'.

At low drive amplitudes, only two steady state motions are possible—near harmonic os-

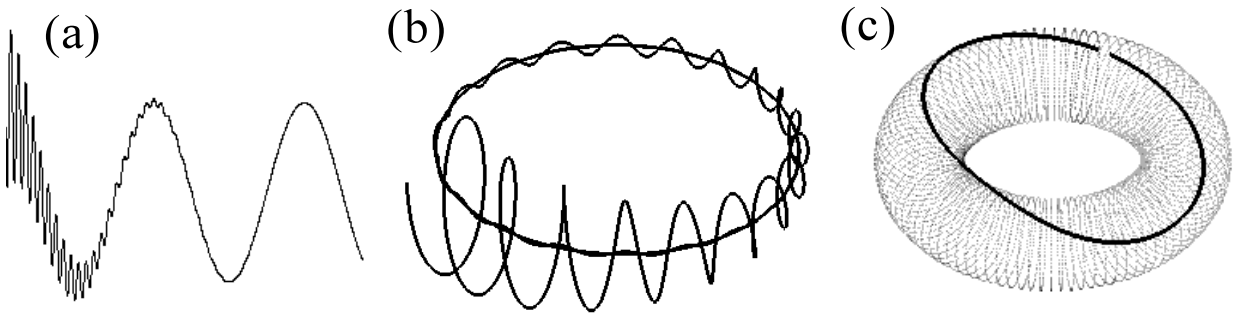

Figure 4: (a) θ vs. t for a linear oscillator. (b) The same data represented as a phase space trajectory. (c) The dark curve represents the steady state trajectory (attractor) for a linear oscillator. It lies on a torus.

cillations centered about one or the other equilibrium point. Each of these motions corresponds to a period-1 limit cycle attractor. An interesting question not relevant to linear oscillators is to determine which of the two attractors the motion will end on. The initial conditions do, in fact, unambiguously determine which attractor the motion will end on. Thus, a *basin of attraction* can be associated with each attractor, and is defined as those phase points that lead to motion on that attractor. For nonlinear systems, the shapes of these basins can be quite complex.

At larger drive amplitudes, chaotic motion can occur for which the trajectory never repeats. The phase point cycles through the full set of θ-ω planes once per drive period yet never hits the same point twice. Moreover, an attractor still describes the ultimate motion. As with the linear oscillator, arbitrary initial conditions cause motion off the attractor, but after this transient motion decays away, the system phase point ultimately moves on a subset of points in phase space. The subset is the attractor, and determining its shape and properties are important parts of the analysis.

One trivial property is that it must lie within a bounded three dimensional volume of phase space. Under most operating conditions the bounds for θ are $|\theta| < 2\pi$ (the pendulum never goes over the top twice in the same direction). The range of ϕ is also bounded; its extents can be taken from 0 to 2π. And ω must be bounded as well. The kinetic energy $I\omega^2/2$ will not grossly exceed the largest possible potential energy $k(r\theta)^2$ with $\theta \approx 2\pi$.

Just as a point in phase space represents the state of a system, a volume in phase space represents a set of states. For every value of (θ, ω, ϕ) in that volume, the relation $\dot{\mathbf{u}} = \mathbf{G}(\mathbf{u})$ provides unique values for the derivatives of each variable. The derivatives specify a unique flow for each trajectory point so that as the trajectories evolve, they can converge toward one another and they can diverge away from one another but they can never cross. A crossing would require two different values for $\dot{\mathbf{u}}$ at the same value of $\mathbf{u}$, forbidden under the assumption that the pendulum is governed by $\dot{\mathbf{u}} = \mathbf{G}(\mathbf{u})$.

As outlined in the appendix on the Lyapunov exponent algorithm, one can prove that any finite volume of phase points simultaneously evolving according to Eq. 25 (with Eq. 28) will exponentially decay to zero with a rate constant given by $-\Gamma$. Thus, consider a set of initial conditions filling a finite vol-

ume of phase space that completely covers the attractor. As the transients decay away, the phase points must all come to lie on the attractor while at the same time the volume of the attractor must exponentially decay to zero. This is possible if the attractor consists of any number of limit cycles because a limit cycle is a finite length curve through phase space and thus has zero volume. However, limit cycles are not the only possibility. In fact, chaotic motion never repeats. Consequently, the attractor on which it lies can not be a limit cycle and must be a curve with an infinite length. Thus, the chaotic attractor must be a set of points with an infinite length but zero volume. How can this be? A two dimensional surface of arbitrary shape in phase space is a set of points with this property, but it will turn out that our attractor is not any kind of oddly shaped area in phase space. You will determine that the dimensionality of our attractors is between that of an area (a two dimensional object) and a volume (a three dimensional object). Chaotic attractors apparently have strange shapes with fractional dimensions (fractals). Fractal attractors are called *strange attractors*.

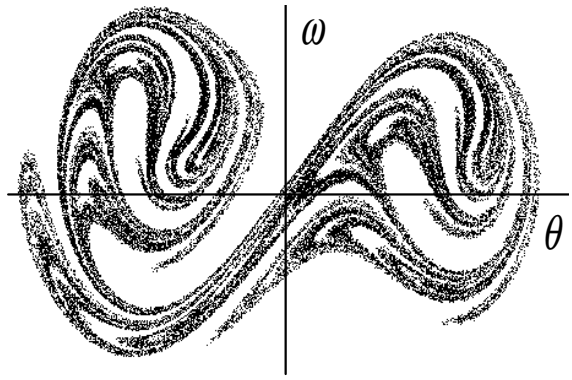

Figure 5: A Poincaré section from the apparatus of Fig. 1.

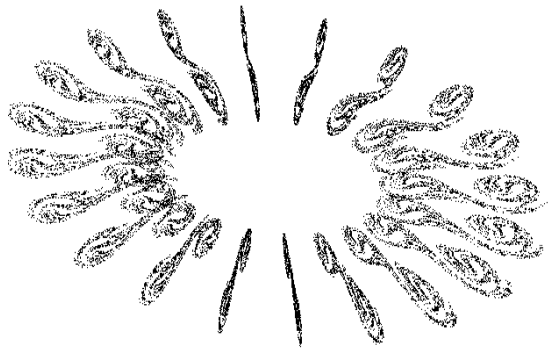

Figure 6: A sequence of Poincaré sections arranged to illustrate the complete three dimensional attractor in the phase space coordinate system of Fig. 3

Poincaré sections

We will only be able to measure a subset of the complete attractor—a very long trajectory. Overnight runs of 50,000 drive cycles are typical for our apparatus. Such a long trajectory is nearly enough to completely represent the attractor and we will often not distinguish between the trajectory and the attractor. However, you should always keep in mind that the full attractor has infinitely more points than any finite trajectory.

A long trajectory is difficult to display on a two-dimensional medium such as a computer screen or a piece of paper. One way to achieve a compact representation of a very long trajectory is with a *Poincaré surface of section* (Poincaré section for short)—so named because it is a cross section of the full three-dimensional trajectory. In a Poincaré section, θ and ω are recorded as a single point each time the trajectory pierces through the θ-ω plane corresponding to one value of the drive phase ϕ. Thus, once per drive period, a new (θ, ω) point is added to the plot as the trajectory repeatedly pierces that particular plane.

A Poincaré section from our apparatus is shown in Fig. 5. A collection of Poincaré sec-

tions at sequential drive phases ϕ is a representation of the complete attractor. Figure 6 shows a three dimensional arrangement of 20 Poincaré sections illustrating the attractor. Several sequences of 40 Poincaré sections can be seen as movie loops on the laboratory web page.

Fractal dimension

The attractor for chaotic motion is called "strange" because it has a strange shape called a fractal. A fractal is a self-similar object, meaning it looks similar on ever decreasing scales. One invariant of a strange attractor is its fractal dimension. Since our attractor is dimensionally the same as an extruded Poincaré section, its fractal dimension is one plus the fractal dimension of any of its Poincaré sections. Consequently, it is the fractal dimension of a Poincaré section that will be sought. While there are several different varieties of fractal dimensions, the most common is the *capacity* dimension. The box-counting algorithm used to determine the capacity dimension also demonstrates the self similarity of a fractal Poincaré section.

The algorithm for a two dimensional fractal is as follows. Determine the bounds in the plane over which the fractal extends. Divide that area into $M \times rM$ grid boxes ($r = 1$ is fine but not required). Count the number of grid boxes N covering any part of the fractal. Repeat the procedure as M is increased, making the grid boxes smaller and smaller. The slope of the $\log N$ vs. $\log M$ graph (in the limit as $M \to \infty$) is the capacity dimension.

The procedure above is for the idealized case of an infinite trajectory—a Poincaré section from the complete attractor. It will need to be modified for an experimental Poincaré section obtained from a finite trajectory. The phrase "covering any part of the fractal" is modified to "having at least one phase point." The phrase "in the limit as $M \to \infty$" must be dropped because if the size of boxes become too small, N will saturate at the total number of phase points in the Poincaré section. Thus, one must simply check that the graph of $\log N$ vs. $\log M$ appears to be a straight line before N saturates. For our 50,000-point Poincaré sections, $r = 1/2$ (twice as many grid lines in θ as in ω) and M in a range from 10 to 300 or so works well.

C.Q. 3 *The ordinary geometric dimensions of a point, line, area, and volume are 0, 1, 2, and 3, respectively. (a) Explain how the box-counting algorithm would give a dimension of 0, 1, and 2, respectively, for a very large number of phase points distributed uniformly at a point, along a line, and over an area in the θ-ω plane. (b) When there are two or more such objects, e.g., points and lines or lines and areas, explain why the algorithm should give the dimension of the highest dimensional object in the set.*

Lyapunov exponents

Fractal attractors are one hallmark of chaotic behavior. Extreme sensitivity to initial conditions is another. It is typified by an exponential growth in the phase-space separation of two trajectories which start out very near one another. Actually, at each point $\mathbf{u}$ on the attractor there are three different directions (global Lyapunov vectors) $\hat{\mathbf{e}}_i$ that characterize three different exponential time dependencies that occur. Any phase point $\mathbf{u}'$ near $\mathbf{u}$ having a small difference

$$\delta\mathbf{u} = \mathbf{u}' - \mathbf{u} \tag{30}$$

that is parallel to one of these directions, evolves with a single exponential time dependence characterized by one of three Lyapunov

exponents. In other words, any initial separation $\delta\mathbf{u}(0)$ in the direction of a particular Lyapunov vector $\hat{\mathbf{e}}_i$ evolves according to

$$|\delta\mathbf{u}(t)| = |\delta\mathbf{u}(0)|e^{\lambda_i t} \tag{31}$$

where λ_i is the corresponding Lyapunov exponent for the particular direction $\hat{\mathbf{e}}_i$.

The λ_i can be positive, negative, or zero. A negative value corresponds to a direction in which the separation decays over time. A zero value corresponds to a direction in which the separation remains constant. And for chaotic motion, at least one of the λ_i must be positive—corresponding to a direction in which the separation grows. No matter how small, any two initial conditions will always have some component along the expanding direction, and the trajectories evolving therefrom will always diverge.

Small random forces and torques acting on the pendulum (e.g., from the pickup of vibrations in the pendulum supports and from small variations in friction), cause small perturbations to the phase space trajectory. These perturbations then also evolve according to the system dynamics and are called dynamical noise. Extremely close phase points, which might take a long time to separate in a noiseless system, do so more quickly in the presence of dynamical noise. A study of the effects of dynamical noise would make an interesting advanced project. However, the dynamical noise in our apparatus is small enough that many predictions made assuming it is negligible will still be valid.

One of the three Lyapunov directions and its corresponding exponent is essentially trivial and easily visualized. Imagine two identical trajectories differing only in that one leads the other by some small time interval δt. For small enough δt, Eq. 25 implies their phases will differ by $\delta\mathbf{u} = \mathbf{G}\,\delta t$, and although $\delta\mathbf{u}$ changes as the two systems evolve (because $\mathbf{G}$ depends on $\mathbf{u}$), there is no average increase or decrease with time. Thus, this "propagation" direction has a corresponding Lyapunov exponent of zero. All dynamical systems governed by an autonomous set of differential equations will have such a direction and at least one Lyapunov exponent will be zero.

Because the motion in the propagation direction is basically trivial, the behavior of nearby points on the same Poincaré section provide all the essential information. The general behavior is illustrated in the two Poincaré sections shown in Fig. 7. Consider the single phase point $\mathbf{u}_n$ marked by the cursor in the section on the left. The highlighted points centered around $\mathbf{u}_n$ are a set of nearest neighbors $\mathbf{u}_j$ found by searching through all points on this Poincaré section and selecting those inside an elliptical region satisfying

$$\left(\frac{\delta\theta_j}{\theta_r}\right)^2 + \left(\frac{\delta\omega_j}{\omega_r}\right)^2 < 1 \tag{32}$$

where $\delta\theta_j = \theta_n - \theta_j$, $\delta\omega_j = \omega_n - \omega_j$. The quantities θ_r and ω_r specify the major axes of the elliptical neighborhood (0.15 rad and 0.30 rad/s, respectively, for the section on the left in Fig. 7).

For the point $\mathbf{u}_n$ and the set of points $\mathbf{u}_j$, a corresponding point $\mathbf{u}_{n+1}$ and set $\mathbf{u}_{j+1}$ are also thereby determined as those points one time step later. These points are shown in the section on the right for a time step τ equal to 1/5th of the drive period. From these two sets are constructed two sets of deviations $\delta\mathbf{u}_n = \mathbf{u}_n - \mathbf{u}_j$ and $\delta\mathbf{u}_{n+1} = \mathbf{u}_{n+1} - \mathbf{u}_{j+1}$ one time step forward.

For small values of $\delta\mathbf{u}$, the transformation between the $\delta\mathbf{u}$ in each neighborhood is predicted to be relatively simple—rotations and/or contractions and/or expansions. While the rotations can not be discerned in Fig. 7, a contracting and expanding direction are evident. (The scaling is the same

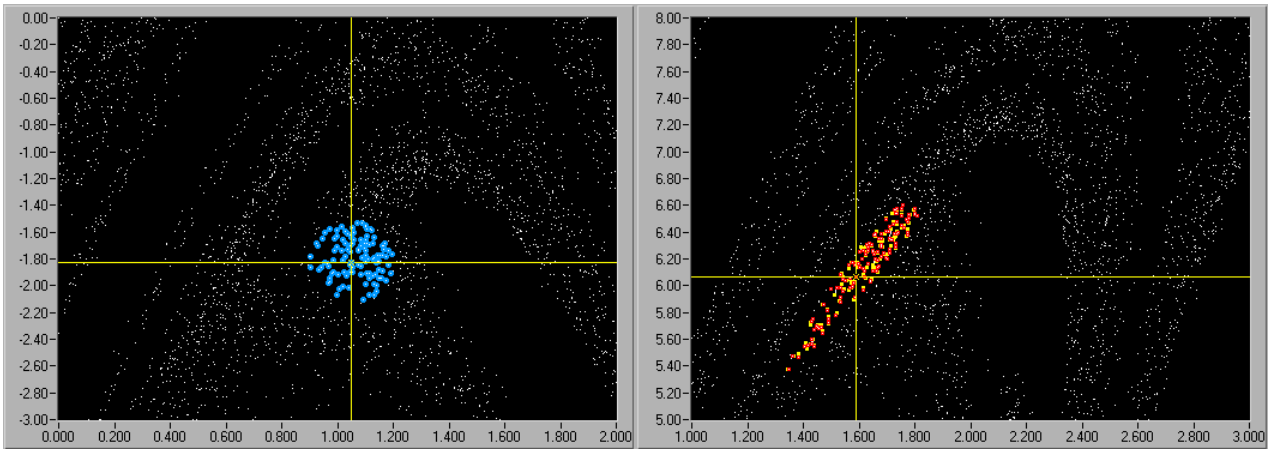

Figure 7: A small area of a Poincaré section showing how an initial set of $\delta\mathbf{u}_n$ around the cursor in the left plot move and change shape when propagated 1/5 period. The scale ranges are the same in the two figures and the evolved set $\delta\mathbf{u}_{n+1}$ is the elongated and flattened patch around the cursor in the right figure.

in the two sections.) The basic transformation can be modeled as a 2×2 matrix multiplication

$$\begin{pmatrix} \delta\theta_{n+1} \\ \delta\omega_{n+1} \end{pmatrix} = \begin{pmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{pmatrix} \begin{pmatrix} \delta\theta_n \\ \delta\omega_n \end{pmatrix} \tag{33}$$

with the matrix elements b_{ij} depending on where $\mathbf{u}$ is located on the attractor. A least squares fit of the data (the $\delta\mathbf{u}_n$ and $\delta\mathbf{u}_{n+1}$) to Eq. 33 provides best estimates of the transformation matrix elements.[5] The matrix elements are then used to determine two *local* Lyapunov exponents which must be averaged along a long trajectory to determine the global Lyapunov exponents characteristic of the attractor as a whole. From an arbitrary starting point, this procedure—finding the nearest neighbors, determining the $\delta\mathbf{u}_n$ and $\delta\mathbf{u}_{n+1}$, performing the fit, and determining the two local Lyapunov exponents—is repeated at sequential points along a trajectory. The local Lyapunov exponents are monitored and the calculation continues until their average is deemed to have stabilized.

A more complete description of the algorithm and its theoretical underpinnings is provided in the Lyapunov Exponent addendum that can be found on the laboratory web site.

Fit of the angular acceleration

The Savitsky-Golay filters can be used to calculate the angular acceleration at every point along a trajectory. Modeling the values of α $(= \dot{\omega})$ according to Eq. 8, a linear regression can be used to experimentally determine Γ, Γ', κ, μ, and ϵ.

For the fit, it is important to take into account the differences between the measured angles and those of the theory. For example, $\theta = \theta_m + \delta\theta$ would be used to describe the fact that the computer value θ_m may be offset from that of the theory θ. Similarly, $\phi = \phi_m + \delta\phi$. Lastly, the possibility that the spring equilibrium may not be set exactly at $\theta = 0$ should be taken into account; it is taken to occur at

[5] As shown in the Appendix on Lyapunov exponents a modified version of Eq. 33 is actually fit to the data.

some unknown angle θ_0.

With these offsets in mind, Eq. 8 becomes

$$\alpha = -\Gamma\omega - \Gamma' \mathrm{sgn}\,\omega - \kappa(\theta_m + \delta\theta - \theta_0) + \mu \sin(\theta_m + \delta\theta) + \epsilon \cos(\phi_m + \delta\phi) \quad (34)$$

This equation can be put in a form suitable for regression analysis using the trigonometric identities: $\sin(a+b) = \sin a \cos b + \sin b \cos a$ and $\cos(a+b) = \cos a \cos b - \sin a \sin b$.

C.Q. 4 *(a) Show that the regression expression can be written:*

$$\begin{aligned} \alpha &= C[1] + G[\omega] + G'[\mathrm{sgn}\,\omega] + K[\theta_m] \\ &\quad + M_s[\sin\theta_m] + M_c[\cos\theta_m] \\ &\quad + E_c[\cos\phi_m] + E_s[\sin\phi_m] \end{aligned} \quad (35)$$

where each regression term is in brackets and where the coefficients of each term are given by

$$\begin{aligned} C &= -\kappa(\delta\theta - \theta_0) \\ G &= -\Gamma \\ G' &= -\Gamma' \\ K &= -\kappa \\ M_s &= \mu \cos\delta\theta \\ M_c &= \mu \sin\delta\theta \\ E_c &= \epsilon \cos\delta\phi \\ E_s &= -\epsilon \sin\delta\phi \end{aligned} \quad (36)$$

(b) After obtaining these coefficients from the fit, they are used to determine the parameters. Show that this becomes:

$$\begin{aligned} \Gamma &= -G \\ \Gamma' &= -G' \\ \kappa &= -K \\ \mu &= \sqrt{M_s^2 + M_c^2} \\ \delta\theta &= \tan^{-1}\frac{M_c}{M_s} \\ \epsilon &= \sqrt{E_c^2 + E_s^2} \\ \delta\phi &= \tan^{-1}\frac{-E_s}{E_c} \\ \theta_0 &= \tan^{-1}\frac{M_c}{M_s} - \frac{C}{K} \end{aligned} \quad (37)$$

While the Savitsky-Golay filters can be quite accurate when fitting continuous functions, the discontinuous behavior of axle friction (it changes sign at $\omega = 0$) presents a slight problem. Because the filters average over a finite time, they smooth the discontinuity over a range that depends on the number of points used in the filter. For the recommended 33-point filters, the axle friction smoothing appears to be over a range of ± 3 rad/s. To prevent systematic errors, the fits should exclude data points in this range.

LabVIEW Programs

The following sections describe the use of the data acquisition and analysis programs available for this experiment. All analysis programs are limited to three-dimensional attractors like those for the system described here. The analyses typically begin by converting the data sets to a discrete set of two-dimensional Poincaré sections equally spaced along the trivial (drive phase) dynamical variable.

The analysis programs are designed to work with several kinds of data sets that can be generated in our lab. In the programs they are labeled:

- **Poincaré:** A text-type spreadsheet file whose first two columns contain a single Poincaré section.
- **I16:** A binary file containing a single row of 16-bit signed integers. This is the correct file type for the chaotic pendulum.
- **datalog:** A binary file containing multiple rows of double precision numbers.

This is the type of file made by the simulation programs.

- **N×I16:** A binary file containing multiple rows of 16-bit signed integers. This is the type of file made when collecting data for a driven electronic circuit. The first two indices in the var. order control then determine which variables are used in the Poincaré sections.

All of the programs are started by clicking on the Run button on the toolbar. Different analysis programs may run with only particular kinds of data sets. The File type tab control on the front panel of the program will have a tab for each allowed type and the type desired must be selected before hitting the Run button. For all but the *DisplayI16* program for real time display of experimental data, various data set parameters (described next) must also be selected before running the program.

For the I16 file type there is a delta t control giving the time between each measured integer. It is used with the Savitsky-Golay filtering to determine properly scaled derivatives. For the chaotic pendulum it should be set to 1/200 of the drive period. For both the I16 and N×I16 file types there is a delta y control (array for the latter) which gives the physical step size for each change of 1 in the 16-bit integer. For the chaotic pendulum, it should be set to $2\pi/1440$, the default setting.

Savitsky-Golay filtering is used to determine θ, $\omega = \dot{\theta}$, and $\alpha = \ddot{\theta}$ values from the raw rotary encoder readings.[6] This filtering is equivalent to performing an unweighted, linear least squares fit of the readings to a polynomial at each point. The control for filter averaging (pts) is used to set the number of consecutive data points used (33 is recommended) and the polynomial order is set by the polynomial order control (a 4th order or quartic polynomial is recommended).

The programs may have from and to controls and/or a Slice control. The from and to controls determine which drive cycles in the data set will be analyzed. For a data set with 50,000 drive cycles, choosing the from and to to be 5001 and 15000 would cause the program to use the ten thousand cycles starting from cycle 5001.

Programs requiring multiple Poincaré sections from the attractor will have a sections/cycle control. These programs use a set of n equally-spaced Poincaré sections, where n is the setting for this control. Programs that use only a single Poincaré section will have an additional slice control to select which of the n Poincaré sections the program will use. Both kinds of programs must also know the total number of data points per drive cycle and will have a readings/cycle control, which should be set for 200 for experimental chaotic data (I16 files) or 40 for simulations (default-sized datalog files).

Acquire chaotic pendulum data

Raw data from the rotary encoder (1440 pulses per revolution) are collected and saved to a file by the *Acquire chaotic pendulum data* program. This program steps the (200 pulse-per-revolution) drive motor at a constant rate and triggers a reading of the rotary encoder with each motor pulse.

The drive frequency (in rotations per second) is set by the stepper motor frequency control. After the Run button in the toolbar is pressed, the control is updated with the nearest frequency the system was able to reach. This program will ask for a file name in which it will store the data. Navigate to the *My Documents* area to store the data. Unless this is a run you are sure you will want to save, keep

[6] See the laboratory chaos web page for addendum on the theory of Savitsky-Golay filters.

the default name Data.bin and click OK when it asks if it can be overwritten. You can always rename it afterward if you decide it will be needed later. The reset button can be pressed while taking data to restart saving data at the beginning of the file. However, previous data remains part of the file until overwritten.

You can change the drive frequency two ways. The automatic method repeatedly changes the frequency by a specific amount while continuously taking data and it should only be used for independent study. At the bottom of the front panel there is a button to turn the frequency sweep feature on and off and a control for the frequency change per step (negative values are OK). Each step is applied after a fixed number of pulses to the stepper motor (step delay). The default value gives a sweep of -0.001 Hz/cycle. With the frequency sweep off, the frequency can be changed manually and "on the fly" while data collection continues. To do so, simply change the value in the stepper motor frequency control and then click on change frequency button. Keep in mind that unless the drive frequency is constant for the entire run, any programs using the delta t control (described later) will not have correct time or frequency scales. All information about the run, such as drive frequency, the magnet damping distance and the drive shaft length, should be written in your lab notebook.

The angular position of the drive shaft and of the rotary encoder when the toolbar Run button is pressed are taken to be $\phi = 0$ and $\theta = 0$, respectively, in the data analysis programs. Consequently, if you want $\phi = 0$ and $\theta = 0$ to be close to the straight up positions, you will have to set the drive shaft and hold the pendulum in these positions as you press the Run button. To set the drive shaft, turn the motor controller off, adjust the shaft by hand to the vertical position, and turn the motor controller back on before pressing the Run button. However, these zeroing procedures are hardly ever relevant. Just keep in mind that in any analysis, the drive shaft and pendulum angle for any data set will generally have an unknown offset. The offsets are irrelevant for most analyses, and when they are relevant, for example in the fits to the angular acceleration, they are included as parameters.

For collecting data on undriven motion, set the drive frequency to 1.0 Hz. This will establish the rotary encoder data collection rate at 200 readings per second. Set the drive shaft for $A = 0$ and shut off the motor controller to prevent the motor from turning.

DisplayI16

Use the *DisplayI16* program to look at the data while it is being collected, to look at short data sets, or to make ASCII files of the data suitable for spreadsheets.

Before hitting the toolbar Run button, enter values for the readings/cycle (200), delta y ($2\pi/1440$) and delta t ($T/200$, where T is the drive period). The history length (cycles) control (described shortly) must also be selected before running the program. After hitting the Run button, the program asks for a data file. Specify the file name given for the *Acquire chaotic pendulum data* program—Data.bin or any previously collected binary data files. However, depending on the file and buffer size, only points from the end of the data set may be available to look at.

The topmost raw readings graph displayed on the front panel is the raw count from the rotary encoder vs. the cycle number. The history length (cycles) control sets the number of drive cycles N that will be available for use by this program. As the data set grows, only the most recent N cycles will be available. The button labeled real time updating should be set

to on to update the buffer with new data as it is being collected. New data is then added to the end of the buffer as old data is removed from the beginning. Click real time updating to off to stop reading in new data and to make it easier to select data for further processing. When you later click it back on, all unread data will then be read in.

Set the cursors on the raw readings graph to cover the desired data and click on the update graphs button to the left of this graph to display the selected data on the other three graphs on the front panel or to save it to a spreadsheet file. The leading cursor determines the first data point processed, while the sections/cycle determine the subsequent data points processed until such point would be after the trailing cursor. (For example, at 40 sections/cycle, every fifth point is processed.) Each of the other three graphs and the save x, v, a to spreadsheet have a button next to them labeled update or don't update. For the graphs, the state of these buttons determines whether the graphed data is updated according to the new settings or left unchanged. For the save $\theta\omega\alpha$ to spreadsheet the update/don't update determines whether the processed data will be saved to a text file readable by most spreadsheet programs. The format %.7e can be changed. Change the 7 to the number of digits to the right of the decimal point.

The graph directly below the raw readings is a plot of x (θ), v (ω), and/or a (α) vs. t, as chosen by the check boxes to the left of the graph. The time interval between points is calculated assuming the raw acquisition rate is constant—200 times the (starting, saved) motor frequency. Thus, if frequency scanning were used, the time scales on these graphs will be incorrect. To the right of the vs. t graphs is the Fourier power spectrum (associated with x vs. t). Below the vs. t graphs is a graph of ω vs. θ. Set the Readings/rev to 1 and the plot style to points/no line to get a Poincaré section. Set the Readings/rev anywhere from 40-200 and change the plot style to line/no points for a phase plot. (To set the plot style click in the legend box at the upper left of the graph. The Common Plots options let you set whether points, lines, or both are displayed.)

Chaotic pendulum fit

The algorithm is described in the theory section. The program needs to be told the data for the fit. You do not want to fit too big a data set. It will bog down the computer. Fitting about ten thousand drive cycles at 10 sections per cycle works well and goes reasonably fast. You can try fitting data from the beginning, middle, and end of the data set (with the from and to controls) to see if the parameter estimates are stable throughout the run. The parameters displayed on the right of the front panel are those of Eq. 34. However, the displayed covariance matrix is that of the fitting parameters of Eq. 35. It is a matrix of the sample covariances s_{ij}^2 (diagonal elements are the square of the estimated parameter uncertainties), with the rows and columns of the matrix in the following order: G, K, M_s, M_c, E_s, E_c, C, G'.

The fit residuals are the differences between α as determined by the Savitsky-Golay filters and the value obtained through Eq. 8 using the fitted parameter values. These can be plot versus θ, ω, or ϕ, and the excluded points can be included in this plot or not.

Box counting dimension

The algorithm is described in the theory section. Recall that the fractal dimension it will find is that of one Poincaré section from the attractor as selected with the Slice control. After clicking Run on the toolbar, the program determines the chosen Poincaré section and then

starts histogramming it. The histogramming consists of setting up a phase space grid and determining the number of measured phase points in each grid box. The overall phase space area it will "grid up" is determined by the minimum and maximum values of θ and ω found in the data. Thus, you should autoscale the graph of the Poincaré section to make sure it does not contain any phase points far from the main attractor which would cause the program to cover an inappropriately large area.

The program repeatedly performs the histogramming as it decreases the grid box size (increases the number of grid boxes). The grids are determined by the array labeled grid size M. For each entry in this array, a phase space grid is formed with M θ-intervals and rM ω-intervals, with the ω-factor (r) as given in the control for this parameter (0.5 is the default). Thus the first default grid ($M = 10$) is a 10×5 grid and thus has 50 grid boxes. The final $M = 400$ value produces a 400×200 grid of 80,000 boxes.

The Probability density shown below the Poincaré section is an intensity graph with the z-value indicated by the color and is proportional to the number of phase points per unit area in each grid box. Looking at this graph, you will see the grid size decrease as the program proceeds.

As each histogram is determined, the cells occupied N indicator displays the number of grid boxes in which at least one phase point was found. This data is plotted on a $\log N$ (ordinate) vs. $\log M$ (abscissa) plot. As described in the theory section, there are good reasons why the value of N for the smallest and largest values of M may not behave as the fractal scaling would predict. Use the cursors to select the starting and ending points of an appropriate region to use in the fit to the straight line. Then click on the compute slope button. The slope of this line is the capacity dimension.

Besides the capacity dimension, two common fractal dimensions are the information dimension and correlation dimension. In all three cases, the dimension is the slope of a fitted line with $\log M$ as the abscissa. For the capacity dimension, each grid box contributes 1 (if there is a phase point in it) or 0 (if it is empty), and the ordinate is the log of the sum of the contributions. For the information dimension, each grid box contributes $f_i \log f_i$, where f_i is the fraction of the phase points in the grid box i, and the graph ordinate is the sum of this quantity over all grid boxes. For the correlation dimension, each grid box contributes f_i^2, and the ordinate is the log of the sum of the contributions from all grid boxes. These other two dimensions can be selected by the Dimension type control.

Lyapunov

The algorithm is described in the theory section and more fully in the Lyapunov Exponent Addendum on the laboratory web site. This program reads the data and makes a set of equally spaced Poincaré sections as determined by the Readings/rev control. (It always uses slice 0 as the first Poincaré section.) Ten sections (i.e., slices) have been found to be enough. No check has been made for the minimum number of drive cycles needed, but 40,000-50,000 is enough to get good results. Use more if you have them, but the computer may start to get bogged down if you try to use too many slices, or too many drive cycles.

The size of the ellipse for selecting $\delta\mathbf{u}_n$ around each point $\mathbf{u}_n$ along the trajectory is determined by the θ-radius and ω-radius controls (θ_r and ω_r in Eq. 32). They should be set to produce a near-circular area and be as small as possible while big enough that the area will nearly always have more than some

minimum number of phase points in it. The program ignores points on the trajectory with fewer available $\delta\mathbf{u}$ than the number set in the Min points control, ten by default. There must be at least six points for the six parameter (quadratically modified) fit of Eq. 33. Having more $\delta\mathbf{u}$ improves the precision of the fitting parameters, but the $\delta\mathbf{u}$ must not get too large. Points on the trajectory are also ignored if the fitted rms deviation exceeds the Tolerance control value. Bad fits have not been a problem and you can leave the Tolerance high to defeat this feature. With the default Order control setting (quadratic) the local fits are as described in the Lyapunov exponent Appendix. Setting it to linear removes the quadratic terms, and setting it to cubic or quartic adds additional terms to the local fit which should not generally be needed. If the higher order fits (having more fitting parameters) are used, the min points control would also need to be increased.)

The operating mode can be set to stationary, single step or iterate. Use stationary to check how the parameters effect the local fit. This mode does not move the calculation along the trajectory so you can try different settings at the same trajectory point. The single step and iterate modes do move the point along the trajectory with the single step mode requiring that you push the step button for each trajectory point, while iterate mode continues along the trajectory without any user intervention until the step button is pressed a second time. Graphing at each point is slow but useful when trying to learn what the program does. The iterations go faster if the graph updating is switched to off once you are ready to determine the Lyapunov exponents.

The Lyapunov contributions $\lambda_i\tau, i = 1, 2$—where τ is the time interval between slices—are displayed for each trajectory point $\mathbf{u}_n$ in the bottom graph. Their sum $(\lambda_1+\lambda_2)\tau$ is also displayed. Set the cursors over the region you would like averaged. You should exclude the first 100 or so cycles of the calculation because the calculation is iterative and improves as the iterations go on. When you click on the average exponents button, the mean and standard deviation of values between the cursors are displayed just below the button. The actual Lyapunov exponents are obtained from these averages by dividing by τ, the time between slices. Thus, if you are using ten slices, multiply by ten to get the Lyapunov exponents per drive cycle (a common way to express them). Further multiply by the drive frequency (in Hz) to get the exponent expressed per unit time as in this writeup.

Lyapunov Dot

This program simply shows a group of $\delta\mathbf{u}$ in a user selected region as they propagate over one or more cycles. Unlike the Lyapunov program, a new set of $\delta\mathbf{u}$ is not redetermined after each time step so you can see how a group of phase points from one small region becomes dispersed throughout a Poincaré section after a certain number of cycles.

Simulate pendulum

This program uses a fourth order Runge-Kutta algorithm to solve Eq. 25. Because the programming environment makes it difficult to use Greek letters, x and v are substituted for θ and ω in this program. Also, the equation $\dot{\phi} = \Omega$ is not used. Instead, the term $\epsilon \sin\phi$ in the equation for $\dot{\omega}$ is simply replaced with $\epsilon \sin \Omega t$.

Clicking on the reset initial conditions button causes the differential equation solver to begin the next drive cycle (at $t = 0$, i.e., $\phi = 0$) with x and v at the values specified in the X and V control just below it.

The points per drive cycle determines the time step. The Runge-Kutta algorithm works best when the time step is neither too big nor too small. Furthermore the points per drive cycle needs to have common divisors if equally-spaced Poincaré sections will later need to be sliced out. The default value of 40 works well. Graphing can be turned off, but for this program, does not significantly affect the running time.

Procedure

Adjusting the springs

The string should be wrapped two and a half times around the middle pulley. When the pendulum mass is installed, the spring equilibrium position should be set at the top center (12:00 position). To make it so, first remove the pendulum mass. I like to do this adjustment with the motor running so that friction is less of a problem. Set the stepper motor shaft for about $A = 1$ cm and run the data acquisition program (described below) at 0.5 Hz. For coarse adjust, hold the aluminum disk and pull down on the string to the left or right of the pulley. For fine adjust, loosen the locking nut and turn the adjustment screw on the cross rod. Retighten the locking nut when finished. When you are satisfied the tap hole is at 12:00, reinstall the pendulum mass.

Preliminary investigations

1. Remove the aluminum disk, determine its mass and diameter and calculate its moment of inertia. This is the major contribution to I_0; the axle and pulley should contribute an additional few percent.

2. Remove the brass pendulum mass and measure its mass m and its diameter. Determine I_m, the moment of inertia of the pendulum mass about its axis. On the aluminum disk, measure the radius l to the tap hole where the pendulum mass will be mounted. Determine the contribution ml^2 to the total moment of inertia. Make a rough estimate of the contribution of the pulley and axle to the total moment of inertia.

3. Measure the diameter of the middle pulley and determine the radius r.

4. Use a spare spring and determine its force constant k.

With the pendulum mass removed, there are a number of experiments that can be performed to learn about the physics of the driven and undriven harmonic oscillator. The following step demonstrates the resonance response of the linear oscillator so that it can be compared later with the response of a nonlinear oscillator.

5. Set up the apparatus for driven motion but leave off the pendulum mass. Set the damping magnet about 5 mm from the disk and set the drive amplitude to about 2.5 cm. Set the drive frequency to 0.2 Hz and start the *Acquire Data* program. Run the *Display I16* program and select this data set. Set the number of sections per cycle to 100 and on the vs. t graphs check only the x data so only the angular position will be displayed when you click on the update graphs button. Don't worry about the delta t control as you will be changing the drive frequency repeatedly — so it won't be accurate anyway. If you leave it at the default ($1/200 = 0.005$), all times will be in units of the drive period and all frequencies will be in units of the drive frequency.

After the transients die out and the motion settles on the steady state oscillations with constant amplitude, turn real time updating off (to hold the display), then set the cursors over the region of constant amplitude motion. Click on the update graphs button to get the x (really θ) vs. t graph. Use the horizontal cursors on this graph to determine the amplitude of the motion in radians.

Turn the real time updating back on and go back to the *Acquire data* program. Change the frequency to 0.25 Hz and click on the change frequency button. As for the previous data, wait for any transients to die out and determine the amplitude of the motion. Continue measuring the amplitude vs. frequency up to around 0.8 Hz or so in 0.05 Hz increments or smaller. As the frequency passes through the natural resonance frequency of the oscillator the amplitude will rise to a maximum and then decrease.

Make a table and graph of amplitude vs. frequency. Determine the frequency where the oscillation amplitude maximizes and determine the FWHM (for the peak amplitudes). Analyze your data with respect to Eq. 22. Compare the resonance frequency with the predicted Ω_0 based on values from the preliminary measurements and determine the damping constant Γ. (This analysis improperly lumps the somewhat weaker effects of axle friction together with the stronger magnetic damping into a single effective Γ.) How would the amplitude versus frequency graph change if the magnet was moved nearer or farther from the disk?

Nonlinear oscillations

6. Adjust the springs as described previously (to get the tapped hole at the top) and install the pendulum mass in that top tapped hole. Don't worry that it will fall to one side or the other when you let it go. Retract the damping magnet, set the drive shaft all the way in (for zero amplitude motion), and turn off the stepper motor controller. Start taking data and then set the disk into motion by hand for oscillations about one of the equilibrium angles. Without stopping the data acquisition, restart the pendulum for motion about the opposite equilibrium angle. Look at the data with the *Display I16* program. Sketch the phase plots. Use the cursors to determine the low-amplitude resonance frequency and the angular separation between the two stable equilibrium points. Compare with predictions.

7. Move the damping magnet closer to the disk—about 5 mm. Again look at oscillations with the drive off and sketch the phase plot. How does it change?

8. In this step you will measure the frequency response of the chaotic pendulum in a manner similar to that used for the linear oscillator. These measurements should all be made with the pendulum oscillating around the equilibrium point on one side only. Adjust the drive shaft for $A = 3$ cm. Start with a frequency of 1.1 Hz — again waiting for transients to die out and then analyzing for the amplitude of the motion using the *Display I16* program. Repeat for frequencies down to 0.3 Hz again in 0.05 Hz increments. Make a table and graph of the amplitude vs. frequency.

9. Repeat the previous step starting at 0.5 Hz and increasing the frequency up to 1.1 Hz or so in steps of 0.05 Hz or smaller. Make a table of amplitude vs. frequency and add it as a plot to the graph from the previous step.

10. Discuss how the amplitude vs. frequency for the linear and nonlinear oscillators differ. In particular, how does the nonlinear oscillator data show hysteresis? Can a linear oscillator have hysteresis effects?

11. Set the frequency to 0.8 Hz, and start data acquisition. At this frequency, you should have obtained large amplitude motion in Step 8 and small amplitude motion in Step 9. Try different initial conditions checking which conditions lead to the larger amplitude motion and which lead to the smaller amplitude motion. Which of the two motions has higher energy? Which has a bigger basin of attraction? Finding complete basins of attraction is not yet practical experimentally. Simply try as many initial conditions as needed (checking which attractor the motion ends on) to decide whether one basin of attraction is smaller, larger, or roughly the same size as the other.

12. Set the drive amplitude to $A = 3$ cm and the drive frequency to $\Omega = 0.5$ Hz. Find all three steady state motions and sketch their θ-ω phase plots. Two are small amplitude oscillations centered about the equilibrium angles $\theta = \pm\theta_e$. The third is a large amplitude oscillation centered about $\theta = 0$. (You will have to give the disk a larger starting amplitude to get this motion.) How many attractors does this system have? Discuss the energy of the oscillations and the basins of attraction.

Limit cycles, Fourier analysis

The periodic motions found above are called period-1 limit cycles. They have Fourier power spectra with peaks at the fundamental Ω and higher harmonics (e.g., 2Ω, 3Ω, ...), but do not have any components below the fundamental. Long-period limit cycles, which repeat after two, three, or more (N) periods of the drive are also possible. You might try to find these motions by playing with the drive frequency, drive amplitude, and/or the damping. A period-N limit cycle will have Fourier spectra with peaks at Ω/N, $2\Omega/N$, ... Peaks at subharmonics of the drive frequency are characteristic of nonlinear oscillators. If you can find a reliable way to get these motions, please let me know. I have spent several hours trying to reproduce one without success. If you do find one, save it and compare its Fourier spectrum with those of period-1 limit cycles.

Chaotic Vibrations

In this part of the experiment, you should try to learn about some of the features of chaotic behavior in nonlinear oscillators.

13. Retract the damping magnet. Set the drive frequency to 0.825 Hz and the drive amplitude for $A \approx 6$ cm. The motion should be chaotic, but if it goes into a limit cycle, decrease the drive amplitude closer to 5 or 5.5 cm. When you get chaotic motion, look at its Fourier spectrum and compare with those for limit cycle behavior.

CHECKPOINT: Procedure should be completed through the prior step.

14. Take an overnight run to obtain a strange attractor for the chaotic pendulum. Repeat with the damping magnet approximately 4 mm from the aluminum disk.

Again, you may need to adjust the drive amplitude to get chaotic motion.

15. Analyze the two attractors for the system parameters using *Chaotic Pendulum Fit*. The default settings should work fine—10 Poincaré sections each with 10,000 points. Discuss the parameters of the two fits. Did the damping parameters behave as expected? How big was θ_0, the spring equilibrium angle? If it is larger than expected, try to determine why.

16. Analyze a Poincaré section or two from each of the two attractors to determine their capacity dimensions using the *Box counting dimensions* program.

17. Analyze the two attractors for their Lyapunov exponents using the *Lyapunov* program. The top two graphs show the group $\mathbf{u}_j$ (left) and $\mathbf{u}_{j+1}$ (right). The bottom graph shows the local contributions for each exponent ($\tau\lambda_i$, red and white dots) and their sum (white line). Setting the two cursors and clicking on the calculate Lyapunov exponents button will display the average and standard deviation of these contributions for the points between the cursors. To get the Lyapunov exponents, you must divide by τ. For example, if you use the default setting of 10 readings/rev, you must divide by $T/10$ (multiply by $10f$) where T is the drive period (f is the drive frequency in Hertz). For $\Gamma' = 0$, the local Lyapunov exponents should sum to $-\Gamma$ everywhere. Compare the sum of the Lyapunov exponents with $-\Gamma$, using Γ from the *Chaotic Pendulum Fit*. Compare the rms deviations of the local Lyapunov exponents with that of their sum. Explain why the latter is so much smaller than former.

18. Simulate the system with the differential equation solver *Simulate pendulum* program. Adjust the differential equation parameters and make simulations comparable with real data sets. Analyze and compare with your measured data for Poincaré sections, fractal dimension and/or Lyapunov exponents.