3

Second Order Differential Equations

We now turn to second order differential equations. Such equations involve the second derivative, $y''(x)$. Let's assume that we can write the equation as

$$y''(x) = F(x, y(x), y'(x)).$$

We would like to solve this equation using Simulink. This is accomplished using two integrators in order to output $y'(x)$ and $y(x)$.

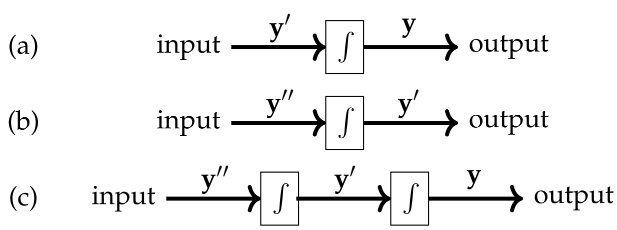

Figure 3.1: Basic schemes for using Integrator blocks for solving second order differential equations.

As shown in Figure 3.1(b), sending $y''(x)$ into the **Integrator** block, we get out $y'(x)$. This is similar to using $y'(x)$ to get $y(x)$ in Figure 3.1(a). As shown in Figure 3.1(c), combining two **Integrator** blocks, we can input $y''(x) = F(x, y, y')$ and get out y and y'. Feeding this output into $F(x, y, y')$, we then obtain a model for solving the second order differential equation. The general schematic for solving an initial value problem of the form $y'' = F(x, y, y')$, $y(0) = y_0$, $y'(0) = v_0$, is shown in Figure 3.2.

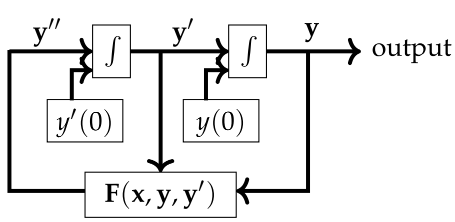

Figure 3.2: This is a general schematic for solving an initial value problem of the form $y'' = F(x, y, y')$, $y(0) = y_0$, $y'(0) = v_0$.

In this chapter we will demonstrate the modeling of second order constant coefficient differential equations and show some simple applications.

3.1 Constant Coefficient Equations

WE CAN SOLVE SECOND ORDER CONSTANT COEFFICIENT DIFFERENTIAL EQUATIONS using a pair of integrators. An example is displayed in Figure 3.3. Here we solve the constant coefficient differential equation

$$ay'' + by' + cy = 0$$

by first rewriting the equation as

$$y'' = F(y, y') = -\frac{b}{a}y' - \frac{c}{a}y.$$

Example 3.1. Model the initial value problem

$$y'' + 5y' + 6y = 0, \quad y(0) = 0, y'(0) = 1,$$

in Simulink.

The simulation in Figure 3.3 solves the equation

$$y'' + 5y' + 6y = 0$$

with appropriate initial conditions. There are two integrators. One integrates the first input, y'', and the other integrates the output of the first integrator, y', giving an output of y. Each **Integrator** block needs an initial condition. The first takes $y'(0) = 1$ and the second needs $y(0) = 0$.

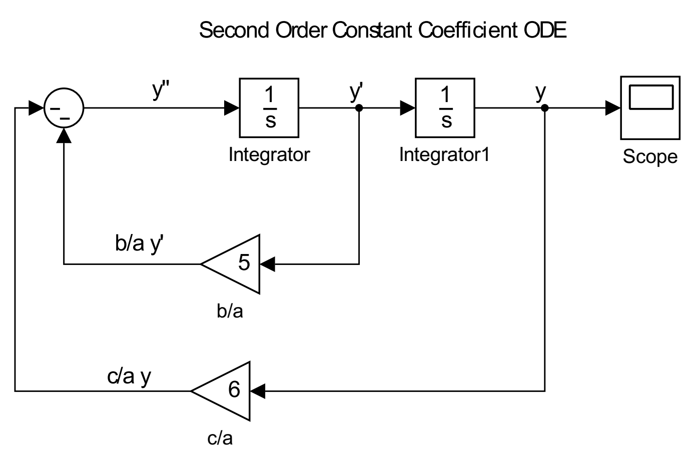

Figure 3.3: Model for the second order constant coefficient ODE $y'' + 5y' + 6y = 0$.

The outputs, y and y' are multiplied by the appropriate constants using a **Gain** block. They are then combined to form the input, $F(y, y') = -5y' - 6y$, to the system of integrators. Running the simulation for 5 units of time, the Scope gives the solution shown in Figure 3.4.

Analytical Solution

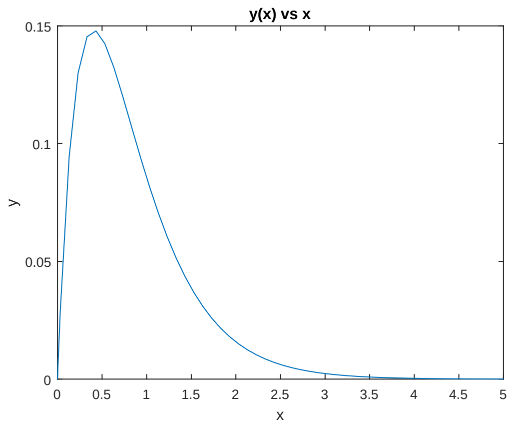

Figure 3.4: Solution plot for the initial value problem $y'' + 5y' + 6y = 0$, $y(0) = 0$, $y'(0) = 1$ using Simulink.

Recall the solution of this problem is found by first seeking the two linearly independent solutions. Assuming solutions of the form $y(x) = e^{rx}$, the characteristic equation is

$$r^2 + 5r + 6 = 0.$$

The roots of the equation are $r = -2, -3$. Therefore, the two linearly independent solutions are $y_1(x) = e^{-2x}$ and $y_2(x) = e^{-3x}$. The general solution is

$$y(x) = c_1 e^{-2x} + c_2 e^{-3x}.$$

The initial conditions hold if

$$0 = c_1 + c_2, \quad 1 = -2c_1 - 3c_2.$$

So, $c_1 = 1$ and $c_2 = -1$. The solution to the initial value problem is

$$y(x) = e^{-2x} - e^{-3x}.$$

The plot of this solution is shown in Figure 3.5. It is seen to agree with the solution shown in Figure 3.4.

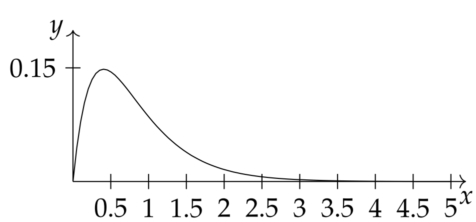

Figure 3.5: Plot of the exact solution of the initial value problem $y'' + 5y' + 6y = 0$, $y(0) = 0$, $y'(0) = 1$.

In the following we will suppress units. In SI units the mass is in kilograms (kg), displacement x is in meters (m), and force is in Newtons (N). Then, k has units of N/m. One could also use CGS units of g, cm, dynes, and dynes/cm, respectively. Time units will generally be in seconds, leaving frequencies in s^{-1}, or Hertz (Hz).

Harmonic Oscillation

A typical application of second order, constant coefficient differential equations is the simple harmonic oscillator as shown in Figure 3.6. Consider a mass, m, attached to a spring with spring constant, k. According to

Hooke's law, a stretched spring will react with a force $F = -kx$, where x is the displacement of the spring from its unstretched equilibrium. The mass experiences a net for and will accelerate according to Newton's Second Law of Motion, $F = ma$. Setting these forces equal and noting that $a = \ddot{x}$, we have

$$m\ddot{x} + kx = 0.$$

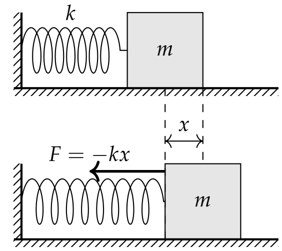

Figure 3.6: A simple harmonic oscillator consists of a mass, m, attached to a spring with spring constant, k.

Here we assume that $x = x(t)$ and let the derivatives be time derivatives. The characteristic equation is given by $mr^2 + k = 0$, or

$$r = \pm i\sqrt{\frac{k}{m}} \equiv \pm i\omega_0.$$

Then, the general solution is given as

$$x(t) = A\cos\omega_0 t + B\sin\omega_0 t.$$

We will model the equation for simple harmonic motion and it variations in the next examples. Namely, we will look at Simulink examples of simple harmonic motion, damped harmonic motion, and forced harmonic motion.

Example 3.2. Simple Harmonic Motion

A Simulink model for simple harmonic motion is shown in Figure 3.7. We write the differential equation in the form

$$\ddot{x} = -\frac{1}{m}(kx).$$

For this example we set $k = 5$ and $m = 2$. We also specify the initial conditions $x(0) = 1$ and $\dot{x}(0) = 0$ in the two integrators.

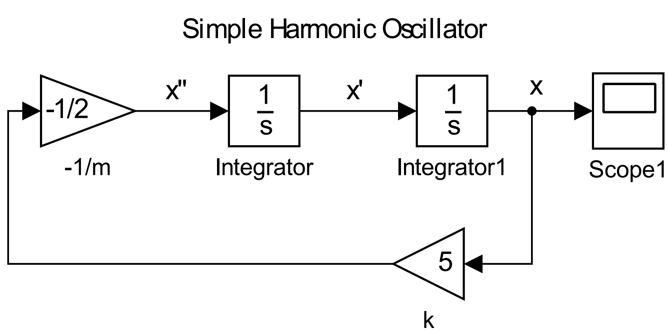

Figure 3.7: A model for simple harmonic motion, $m\ddot{x} + kx = 0$.

The output on the scope is shown in Figure 3.8 for $t \in [0, 10]$. Solving the initial value problem we find that $x(t) = \cos\omega_0 t$, where

$$\omega_0 = \sqrt{\frac{k}{m}} = \sqrt{\frac{5}{2}}.$$

Thus, the period is

$$T = \frac{2\pi}{\omega_0} \approx 3.9738\text{s}.$$

From Figure 3.8 we might have estimated the period as 4 s.

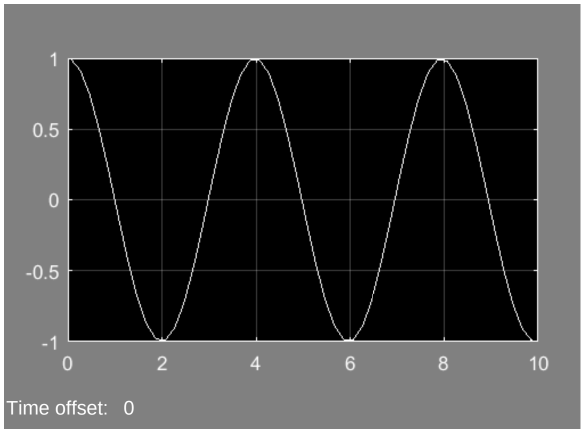

Figure 3.8: Output for the solution of the simple harmonic oscillator model.

Example 3.3. Damped Simple Harmonic Motion

A simple modification of the harmonic oscillator is obtained by adding a damping term proportional to the velocity, $\dot{x}$. This results in the differential equation

$$m\ddot{x} + b\dot{x} + kx = 0,$$

where $b > 0$ is the damping constant.

We can verify the damping behavior in the solution by studying the characteristic equation,

$$mr^2 + br + k = 0,$$

where $x(t) = e^{rt}$ is a guess for form of the the linearly independent solutions. The solutions of the characteristic equation are found using the quadratic formula,

$$r = \frac{-b \pm \sqrt{b^2 - 4km}}{2m}.$$

If $b^2 - 4km < 0$, then the roots of the characteristic equation are complex conjugate roots and the solution takes the form

$$x(t) = e^{-bt/2m}\left[A\cos\omega_0 t + B\sin\omega_0 t\right],$$

where

$$\omega_0 = \frac{\sqrt{4km - b^2}}{2m}.$$

In this case one has oscillatroy solutions with an exponentially decaying amplitude.

A Simulink model for the damped harmonic oscillator can be created using the differential equation in the form $\ddot{x} = -\frac{1}{m}(b\dot{x} + kx)$. This leads to a modification of the model in Figure 3.7. We simply add a term $b\dot{x}$. The model is shown in Figure 3.9.

We consider a specific example using $k = 5$, $m = 2$, and $b = 0.1$. The initial conditions $x(0) = 1$ and $\dot{x}(0) = 0$ are used in the two

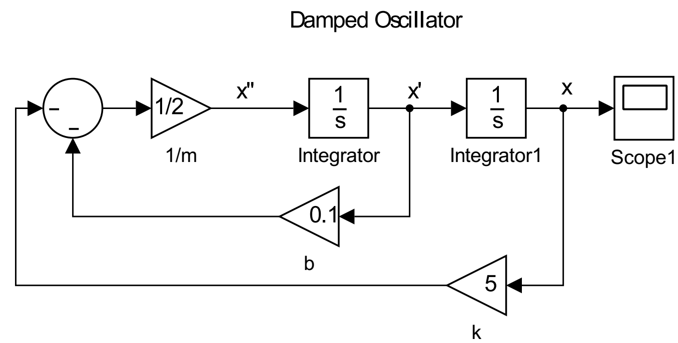

Figure 3.9: A model for damped simple harmonic motion, $m\ddot{x} + b\dot{x} + kx = 0$.

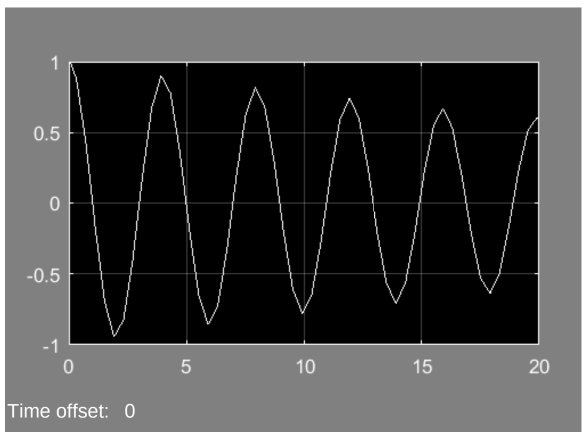

Figure 3.10: Output for the solution of the damped harmonic oscillator model.

integrators. Running the model for $t \in [0, 20]$, the solution seen in the **Scope** block is shown in Figure 3.10. We note that $\omega_0 = 1.5809$ Hz, or the period of oscillation is $T = 3.9743$s. This is consistent with the Simulink solution.

Applying the initial conditions, $x(0) = 1$ and $\dot{x}(0) = 0$, to the general solution, we find that $A = 1$ and

$$\begin{aligned} 0 &= -\frac{b}{2m}A + \omega_0 B, \text{or} \\ B &= \frac{b}{2m\omega_0}. \end{aligned} \tag{3.1}$$

Therefore, the particular solution of the initial value problem can be written as

$$x(t) = e^{-bt/2m}\left[\cos\omega_0 t + \frac{b}{2m\omega_0}\sin\omega_0 t\right].$$

For the parameter values in the problem a plot of this oslution is shown in Figure 3.11 and agrees with Figure 3.10 for this example.

The plot in Figure 3.11 was obtained using MATLAB's **ezplot** function and it symbolic capability. The code is given below for this example.

```
syms t
```

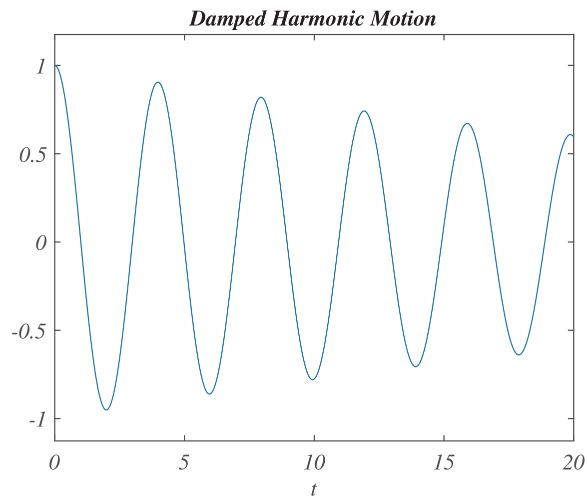

Figure 3.11: The analytic solution for the damped harmonic oscillator example.

```
b=.1; m=2; k=5;
omega=sqrt(4*k*m-b^2)/2/m;
alpha=b/2/m;
A=1;
B=b/(2*m*omega);
x=exp(-alpha*t)*(A*cos(omega*t)+B*sin(omega*t));

ezplot(x,[0,20]);
title('Damped Harmonic Motion')
```

Another modification of the problem is to introduce forcing. In general, the corresponding nonhomogeneous equation is $m\ddot{x} + b\dot{x} + kx = f(t)$. One need only add $f(t)$ to the sum that is sent into the first **Integrator** block. This also requires the **Clock** block and some function blocks. We show this in the next examples.

Example 3.4. Forced Simple Harmonic Motion

We consider a simple sinusoidal forcing and no damping given by

$$m\ddot{x} + kx = F_0 \sin \omega t.$$

The Simulink model in Figure 3.9 is modified to produce the model in Figure 3.12 by adding a **Sine Wave Function** and a **Clock**. We left the damping **Gain** block but set the multiplier to zero. We also note that the **Sum** block shape was changed to rectangular to accommodate more inputs and to direct a consistent flow of the processes.

Using the constants $m = 2$, $k = 10$, we set $F_0 = 1$ and $\omega = 2$ in the **Sine Wave Function**. This results in the output shown in Figure 3.13. Note that the solution is a modulated oscillation. This is understood from looking at the analytic form of the solution.

Recall that we can obtain the analytic solution to this problem using the Method of Undetermined Coefficients. The general solution

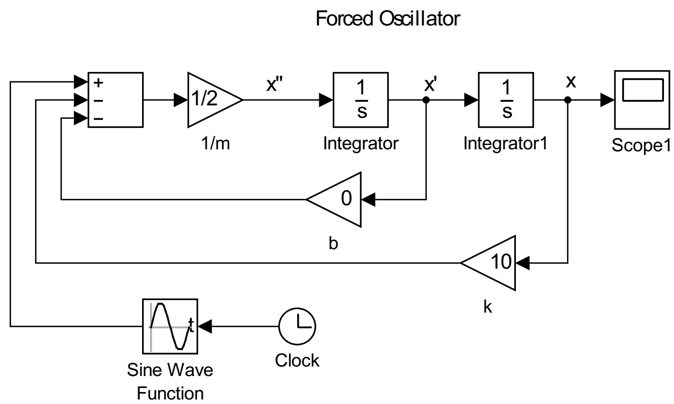

Figure 3.12: A model for forced simple harmonic motion, $m\ddot{x} + kx = \sin \omega t$.

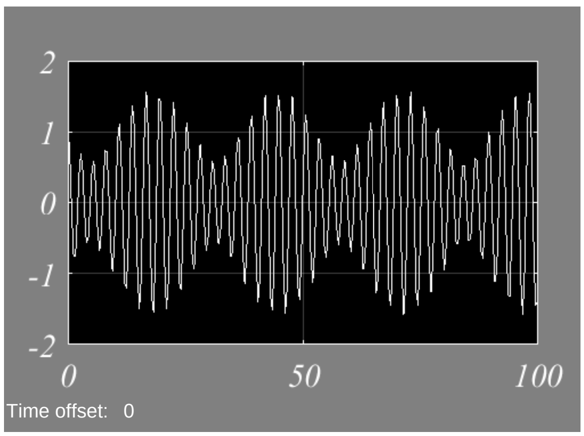

Figure 3.13: Output for the solution of the forced simple harmonic oscillator model.

is a solution of the homogeneous problem plus a particular solution, or guess, to the nonhomogeneous problem. Thus, we have

$$x(t) = A\cos\omega_0 t + B\sin\omega_0 t + x_p(t).$$

We make an educated guess for a function $x_p(t)$ satisfying

$$m\ddot{x}_p + kx_p = F_0 \sin\omega t.$$

Knowing that two derivatives of a sine function returns a constant times the sine function, we assume that $x_p(t) = a\sin\omega t$, providing that this is not a solution of the homogeneous problem. Namely, $\omega \neq \omega_0$.

Inserting this guess into the differential equation, we have

$$-m\omega^2 a\sin\omega t + ka\sin\omega t = F_0\sin\omega t.$$

Since this is true for all t, $-m\omega^2 a + ka = F_0$. Noting that $k = m\omega_0^2$, we can solve for a,

$$a = \frac{F_0}{m(\omega_0^2 - \omega^2)}.$$

Then, the general solution is given by

$$x(t) = A\cos\omega_0 t + B\sin\omega_0 t + \frac{F_0}{m(\omega_0^2 - \omega^2)}\sin\omega t, \quad \omega \neq \omega_0.$$

The initial conditions, $x(0) = 1$ and $\dot{x}(0) = 0$, were again used in the two integrators. The first condition gives $A = 1$. The second condition can be written as

$$0 = \omega_0 B + \frac{F_0 \omega}{m(\omega_0^2 - \omega^2)}.$$

Solving for B, we obtain

$$B = -\frac{F_0 \omega}{m\omega_0(\omega_0^2 - \omega^2)}.$$

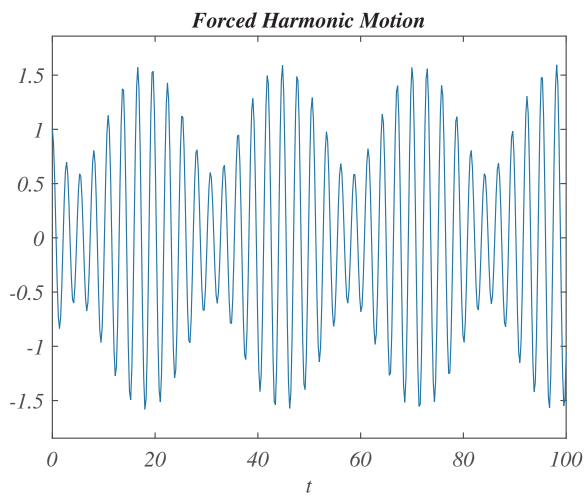

Figure 3.14: The analytic solution for the forced harmonic oscillator example.

Inserting the constants[1] in this problem, the exact solution to the initial value problem is found as

$$x(t) = \frac{1}{2}\sin 2t + \cos\sqrt{5}t - \frac{1}{\sqrt{5}}\sin\sqrt{5}t.$$

[1] Recall that $m = 2$, $k = 10$, $F_0 = 1$, $\omega = 2$. Therefore, $\omega_0 = \sqrt{k/m} = \sqrt{5}$.

The plot of this solution is in Figure 3.14. It agrees with that given by the Simulink model in Figure 3.13.

Example 3.5. Derive a modulation form of the solution from Example 3.4.

Details showing how to derive a modulated form, $x(t) = C(\psi(t))\sin(\theta(t) + \delta)$.

The solution,

$$x(t) = \frac{1}{2}\sin 2t + \cos\sqrt{5}t - \frac{1}{\sqrt{5}}\sin\sqrt{5}t, \tag{3.2}$$

in Figure 3.14 looks like what one would get when adding sinusoidal functions with frequencies that are close. It is the principle used by piano tuners when using a tuning fork to tune a piano key. If the piano key note is slightly different from that of a tuning fork, then when both are sounded at the same time, one hears a beat pattern. This is heard as the low frequency of the envelope similar to that in Figure 3.14. In the last example we had two frequencies, $\omega = 2$ and $\omega_0 = \sqrt{5} \approx 2.2361$, which were close together.

We will combine the the trigonometric functions in Equation (3.2) and show the root of this modulation. We seek a solution in the form

$$x(t) = C(\psi(t))\sin(\theta(t) + \delta),$$

where $C(\psi(t))$ is the modulation amplitude for a higher frequency sinusoidal function and δ is a phase shift. This is accomplished using trigonometric identities.

In the following we will need the result that

$$\begin{aligned} y &= \alpha\cos\theta + \beta\sin\theta \\ &= a\sin(\theta + \varphi). \end{aligned} \tag{3.3}$$

Expanding the second expression, we have

$$a\sin(\theta + \varphi) = a\sin\varphi\cos\theta + a\cos\varphi\sin\theta.$$

Equating coefficients of $\cos\theta$ and $\sin\theta$, we have

$$\alpha = a\sin\varphi, \quad \beta = a\cos\varphi.$$

Adding the squares of these equations,

$$a^2 = \alpha^2 + \beta^2,$$

and taking the ratio of the equations yield

$$\tan\varphi = \frac{\alpha}{\beta}.$$

We now use this result to combine the terms in $x(t)$ into a single sine function with a varying amplitude. We begin by combining the last two terms of Equation (3.2) as

$$\cos\sqrt{5}t - \frac{1}{\sqrt{5}}\sin\sqrt{5}t = a\sin(\sqrt{5}t + \varphi)$$

and set $\theta = \sqrt{5}t$, $\alpha = 1$ and $\beta = -\frac{1}{\sqrt{5}}$. Then, we have that

$$a^2 = 1 + \frac{1}{5} = \frac{6}{5}$$

and

$$\tan\varphi = -\sqrt{5}.$$

Since the angle is in the second quadrant of the $\beta\alpha$-plane, $\varphi = \pi - \tan^{-1}(\sqrt{5})$. This gives the solution in the new form

$$x(t) = \frac{1}{2}\sin 2t + \sqrt{\frac{6}{5}}\sin(\sqrt{5}t + \varphi). \tag{3.4}$$

We now combine the terms in Equation (3.4). Assume that the solution is the sum of the two sine functions

$$x(t) = A\sin(\theta + \psi) + B\sin(\theta - \psi), \tag{3.5}$$

where the variables A, B, θ and ψ are to be determined. It is easy to see that $A = \frac{1}{2}$, $B = a = \sqrt{\frac{6}{5}}$, and

$$\theta + \psi = 2t, \qquad \theta - \psi = \sqrt{5}t + \varphi.$$

Solving this system,

$$\theta = \frac{(2+\sqrt{5})t + \varphi}{2}, \qquad \psi = \frac{(2-\sqrt{5})t - \varphi}{2}.$$

Expanding the sine functions in Equation (3.5), we have

$$\begin{aligned} x(t) &= (A+B)\sin\theta\cos\psi + (A-B)\cos\theta\sin\psi \\ &= [(A-B)\sin\psi]\cos\theta + [(A+B)\cos\psi]\sin\theta \\ &= \alpha\cos\theta + \beta\sin\theta, \end{aligned} \tag{3.6}$$

where

$$\begin{aligned} \alpha &= (A-B)\sin\psi \\ \beta &= (A+B)\cos\psi. \end{aligned}$$

We can combine the terms in $\alpha\cos\theta + \beta\sin\theta$ in the form

$$x(t) = \alpha\cos\theta + \beta\sin\theta = C(\psi(t))\sin(\theta(t) + \delta)$$

using the previous derivation, leading to

$$\begin{aligned} C^2 &= \alpha^2 + \beta^2 \\ &= (A-B)^2\sin^2\psi + (A+B)^2\cos^2\psi \\ &= A^2 + B^2 + 2AB\cos 2\psi \\ &= \frac{29}{20} + \sqrt{\frac{6}{5}}\cos 2\psi \\ \tan\delta &= \frac{\alpha}{\beta}. \\ &= \left(\frac{A-B}{A+B}\right)\tan\psi. \\ &= \left(\frac{\frac{1}{2} - \sqrt{\frac{6}{5}}}{\frac{1}{2} + \sqrt{\frac{6}{5}}}\right)\tan\psi \\ &= \left(\frac{\sqrt{5} - 2\sqrt{6}}{\sqrt{5} + 2\sqrt{6}}\right)\tan\psi. \end{aligned}$$

Thus, we have $x(t) = C\sin(\theta(t) + \delta)$ for C and δ defined by the above relations,

$$\theta = \frac{(2+\sqrt{5})t + \varphi}{2}, \qquad \psi = \frac{(2-\sqrt{5})t - \varphi}{2},$$

and $\tan\varphi = -\sqrt{5}$. This gives a modulated solution by an amplitude envelope with a slowly varying frequency and high freqency oscillations given by the function $\sin(\theta(t) + \delta)$, whose period is $T = \frac{2\pi}{\omega_\theta} = \frac{4\pi}{2+\sqrt{5}} = 2.9665\text{s}$ as compared to $\frac{\pi}{\omega_\psi} = \frac{2\pi}{|2-\sqrt{5}|} = 26.6160\text{s}$ for the envelope. This function is shown in Figure 3.15.

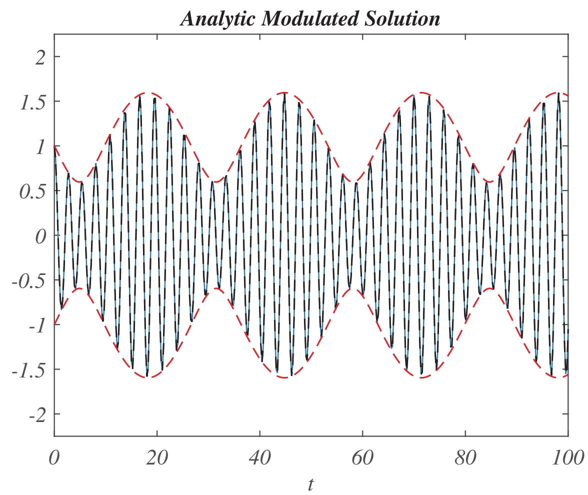

Figure 3.15: The solution, $x(t) = C\sin(\theta(t)+\delta)$, for the forced harmonic oscillator example.

Forced damped oscillator

Example 3.6. Model the forced, damped harmonic oscillator.

A simple application is the forced, damped harmonic oscillator. Recall that this is modeled using a second order, constant coefficient equation,

$$mx'' + cx' + kx = F(t)$$

for some driving force $F(t)$. Rewriting the equation, we have

$$x'' = \frac{1}{m}F(t) - \frac{c}{m}x' - \frac{k}{m}x.$$

This suggests a model like that shown in Figure 3.16. In this example the forcing term was taken as a step function.

$$F(t) = \begin{cases} 0, & t < 1, \\ 1, & t \geq 1. \end{cases}$$

The **Step** function block is set to start at $F = 0$ and increases to a constant value of $F = 1$ after $t = 1$. The other constants are given as $m = 1.0$ kg, $c = 0.5$ kg/s, and $k = 2.0$ N/m.

In Figure 3.17 is shown the solution plot for the forced, damped, harmonic oscillator model with initial values of $x(0) = 1$ and $x'(0) = 0$. In this model there is also an **XY Graph** block. The position and velocity data is fed into this block and the output is a plot of the solution in the phase plane. This is shown in Figure 3.18.

3.2 Projectile Motion

Another example is that of projectile motion. This is a system of equations or a single equation for a vector function. Let the position vector for the projectile be given by $\mathbf{r} = [x, y]$. Then, the projectile satisfies

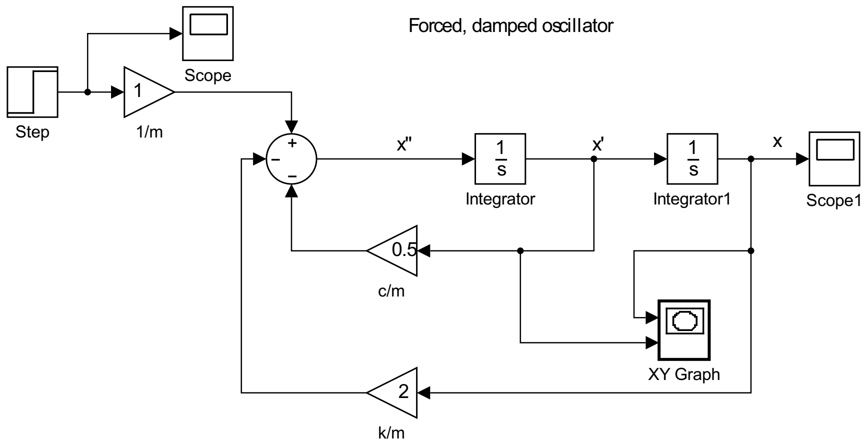

Figure 3.16: Forced, damped oscillator.

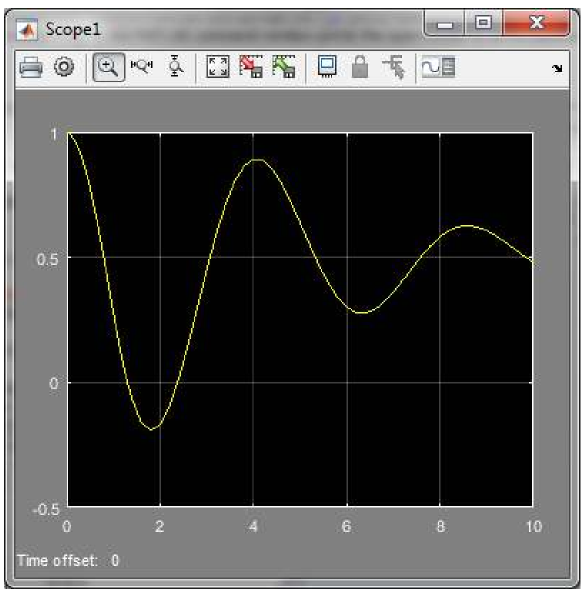

Figure 3.17: **Scope plot** of the solution of the forced, damped, harmonic oscillator model.

the second order equation $\mathbf{r}'' = -\mathbf{g}$. We solve this equation using two integrators and setting up the system with a two component vector. [For a simpler model solving for a two component vector, look at the Bouncing Ball Problem in the next section.]

To make things more interesting, we can add a drag force. Thus, we solve the system

$$\mathbf{r}'' = -\mathbf{g} - kv\mathbf{v}.$$

The magnitude of the drag is proportional to v^2. If the projectile is moving directly upward, the drag is negative, opposing the motion. The model will need functions to compute the speed, v, and will need two integrators with the appropriate initial position and velocity. The gravitational force will also be provided with a constant block. This model is shown in Figure 3.19

For a change, we set up the model in British units (foot-pound-second). The initial position is $[0, 4]$ ft and the initial velocity is $[80, 80]$ ft/s. The

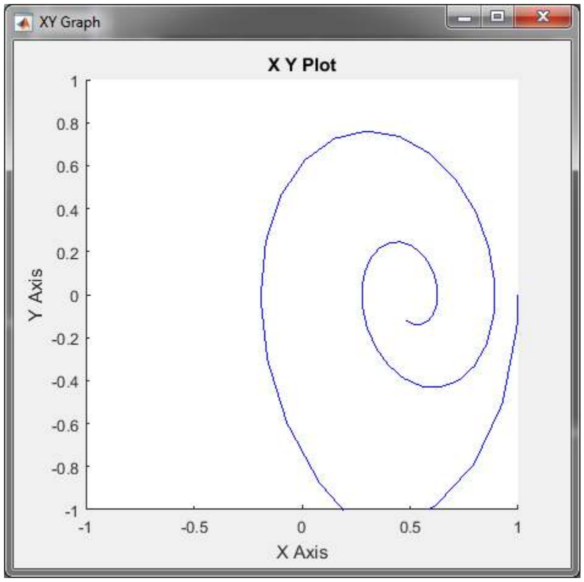

Figure 3.18: **XY Graph** output for the solution of the forced, damped, harmonic oscillator model.

gravitational constant is $-\mathbf{g} = [0, -32]$ ft/s^2. The value of the drag coefficient does not show in the figure. [The value shows when the Gain block is resized.] The position and speed vs time plots are shown in Figure 3.20. Note that changing the simulation time is one way to only display the time that the mass is above $y = 0$. Also, the plot of speed shows that the speed is always positive.

Also shown in this model is the use of the **XY Graph** block. It takes two inputs in order to plot the path y vs x. XY Graphs automatically plot when the simulation is run, as opposed to the **Scope** plots, which need to be double-clicked to display the plots. One also needs to double-click the **XY Graph** block to change the scale shown. For this model the output is shown in Figure 3.21. This plot is useful for determining the maximum height and range of the projectile.

.

3.3 The Bouncing Ball

AS SEEN IN THE PROJECTILE MOTION MODEL OUTPUT in Figure 3.20, the projectile may not stop when it reaches the ground. One needs a way to determine when this has happened and reverse the direction of the motion. In this section we will look at a simple model in which a ball goes through free fall and bounces when it reaches the ground.

The ball satisfies the second order equation $x'' = -g$. Noting that the velocity is $v = x'$, this can be written as two first order equations,

$$\begin{aligned} x' &= v, \\ v' &= -g. \end{aligned} \tag{3.7}$$

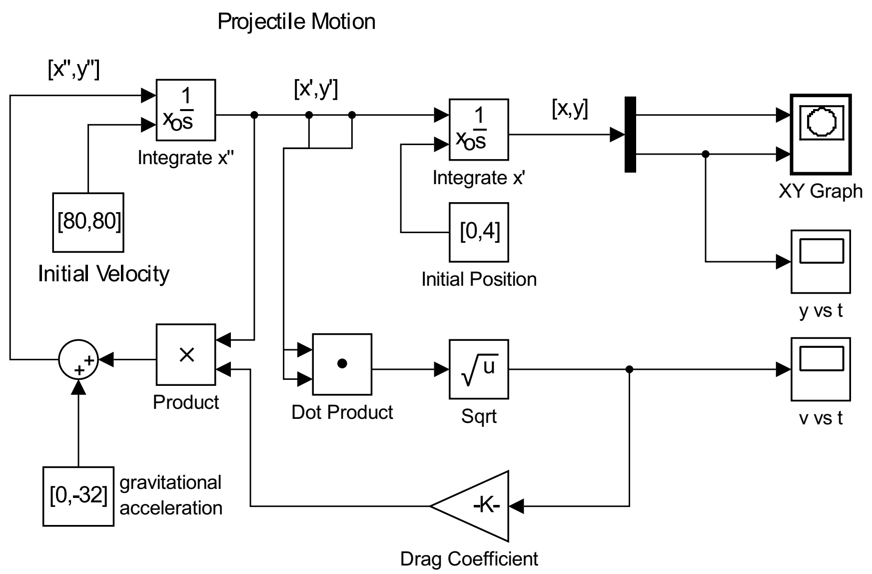

Figure 3.19: Projectile motion model.

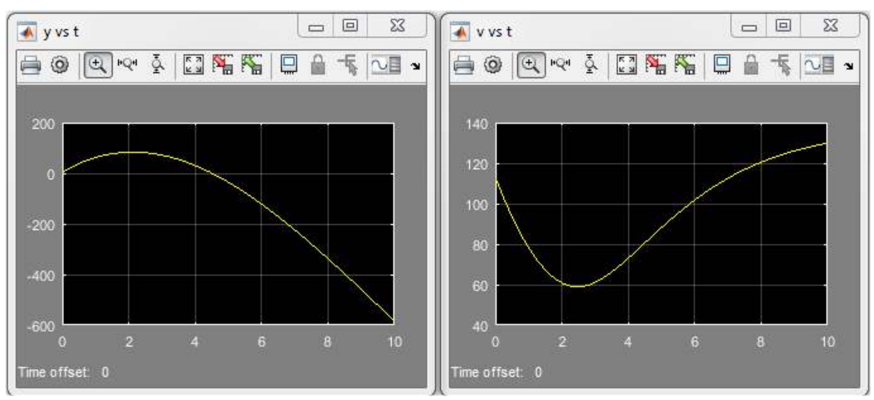

Figure 3.20: Output of the **Scope** Blocks for the projectile motion model for position and velocity vs time.

This system of equations can be then be put into matrix form,

$$\frac{d}{dt}\begin{bmatrix} x \\ v \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} x \\ v \end{bmatrix} + \begin{bmatrix} 0 \\ -g \end{bmatrix}$$

This system can be used to produce the Simulink model in Figure 3.22, where we have introduced initial conditions $x(0) = 3$ and $v(0) = 0$ in the form of an initial vector, $[3;0]$.

In the model we use matrix multiplication to set up the right hand side of the equation. A 2×2 matrix is entered in the gain and the acceleration term is added separately. In order to plot the position vs time, we put a **Demux** block to separate out the components of the "state" vector and added a **Terminator** block to terminate the unused v-branch.

The output of the simulation, which was run for a time of 1 second, is shown in Figure 3.23. Note that the ball has fallen below ground level. We would like for the ball to bounce from the ground. In order to do so, we will test to see when $x \leq 0$ and $v \leq 0$. This is accomplished by adding some test conditions to the **Integrator** block.

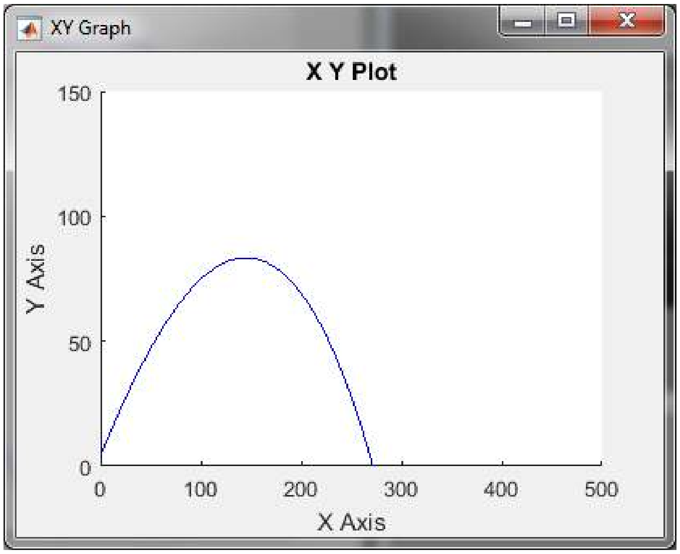

Figure 3.21: Output of the **XY Graph** block for the projectile motion model showing vertical position vs horizontal position.

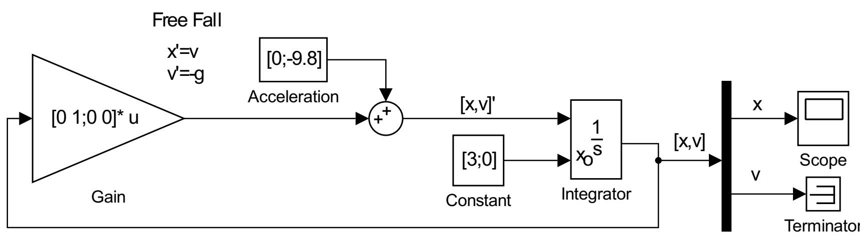

Figure 3.22: Free fall model.

Double-click the **Integrator** block and set the **External reset** to **rising**. This will add a third input as shown in Figure 3.24. Then, replace the initial condition **Constant** block with an **IC** block. This is found in the Signal Attributes group. It looks like the **IC** block in Figure 3.24.

Next, we enter the conditions determining when the block hits the ground and change the block velocity. The input to the condition consist of the Boolean condition, **(u[1]<=0)&&(u[2]<0)**, and the new position and velocity. Here u[1] and u[2] are the position and velocity components. We set the position as **u[1]** and the velocity as **-0.8*u[2]**. These expressions are entered using **Fcn** blocks from the User-Defined Function group. This model is shown in Figure 3.25 with the needed connections to the **Fcn** blocks and the **Integrator** block. This output is shown in Figure 3.26.

3.4 Nonlinear Pendulum Animation

PLOTTING AND ANIMATING SOLUTIONS FROM A MODEL can be done by sending the output of a model to MATLAB. In this section we will solve a nonlinear pendulum problem and show how one sends the output to create a simple animation of the pendulum motion.

A simple pendulum consists of a point mass m attached to a string of length L as shown in Figure 3.27. It is released from an angle θ_0. Newton's

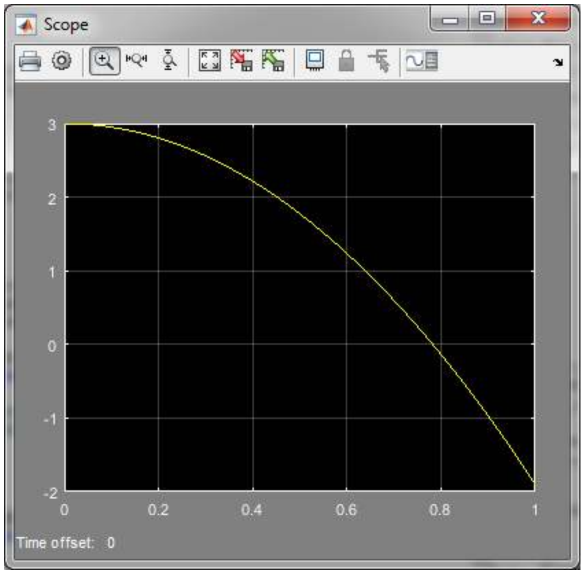

Figure 3.23: Height vs time Scope plot for the free fall model.

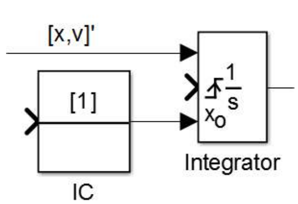

Figure 3.24: Modified **Integrator** and **IC** blocks.

Second Law of Motion tells us that the net force is the mass times the acceleration. So, we can write

$$m\ddot{x} = -mg\sin\theta.$$

Next, we need to relate x and θ. x is the distance traveled, which is the length of the arc traced out by the point mass. The arclength is related to the angle, provided the angle is measure in radians. Namely, $x = r\theta$ for $r = L$. Thus, we can write

$$mL\ddot{\theta} = -mg\sin\theta.$$

Canceling the masses, this then gives us the nonlinear pendulum equation

Nonlinear pendulum equation.

$$L\ddot{\theta} + g\sin\theta = 0. \tag{3.8}$$

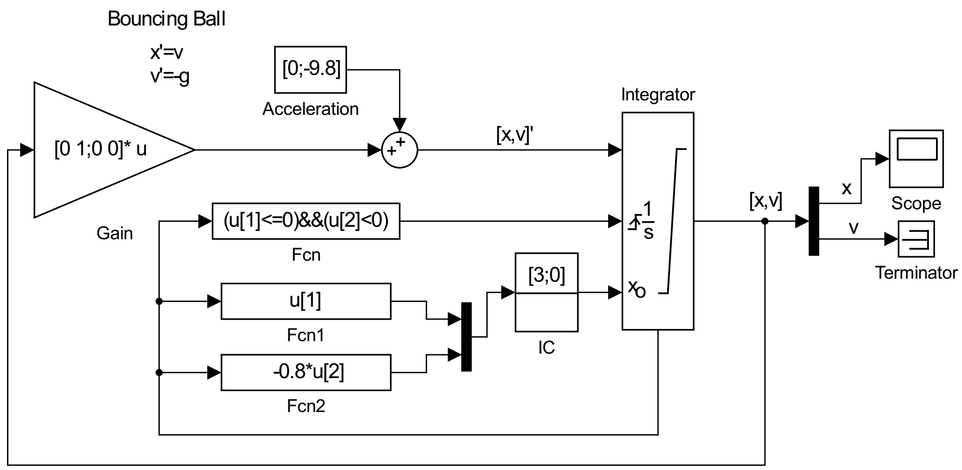

Figure 3.25: The bouncing ball model.

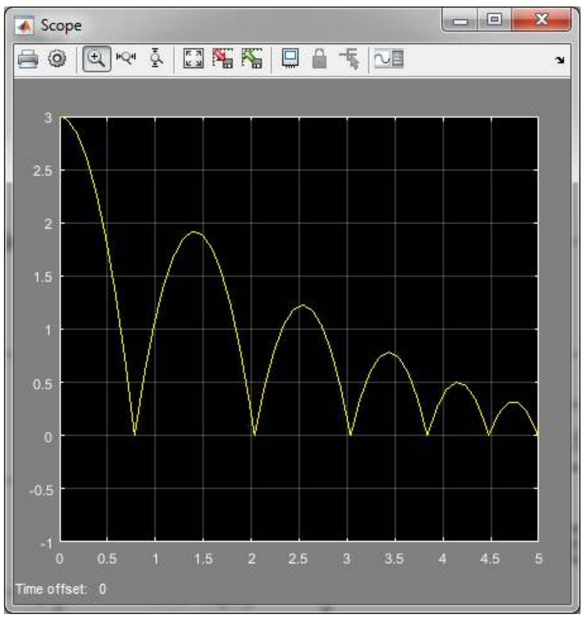

Figure 3.26: Output from the bouncing ball model showing plot of height vs time.

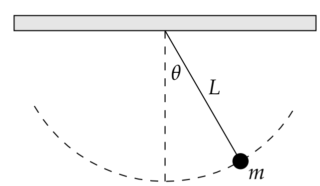

Figure 3.27: A simple pendulum consists of a point mass m attached to a string of length L. It is released from an angle θ_0.

We can use Simulink to model this equation. Such a model is shown in Figure 3.28.It is set up to solve the model in the form

$$\ddot{\theta} = -\frac{g}{L}\sin\theta.$$

The constants are entered using **Constant** blocks and two **Integrator** blocks are used.

We enter the parameters in the system using variables instead of particular constants. These parameters are introduced in a MATLAB m-file. The constants are **L**, **g**, and initial conditions **thetao** and **vo** in the **Integrator** blocks. Save this model as **pend.mdl**.

Now, one creates an m-file, **pendulum.m** with the following code. Here we define the constants first. Then enter the initial conditions followed by the simulation.

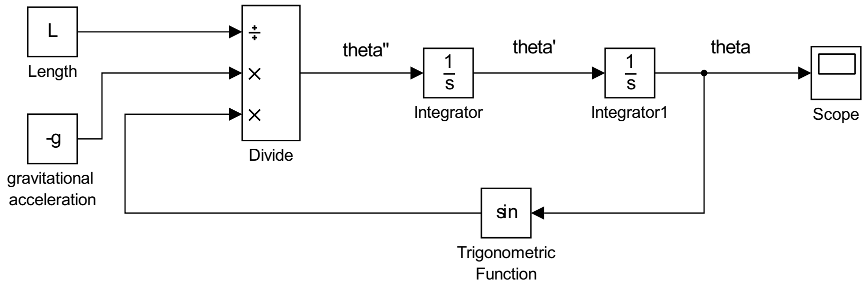

Figure 3.28: Nonlinear pendulum model.

```
m=1.0;
L=1.0;
g=9.8;

v0=0;
theta0=pi/6;
t0=0;
tf=15;

myopts = simset('MaxStep', 0.01);
sim('pend', [t0 tf],myopts)
```

Typing **pendulum** in the command window, assuming that this file and the model are save and run from the same folder, will produce a **Scope** plot for $t \in [0, 15]$. The function **simset** will make the plot smoother.

In order to plot the solution in MATLAB, the solution needs to be output to the MATLAB workspace. This is accomplished by adding a To Workspace block for the **theta** output variable and one for time, using a **Clock**. Double-clicking each block, one can change the output variable names to **theta** and **time**, respectively. The resulting model is shown in Figure 3.29

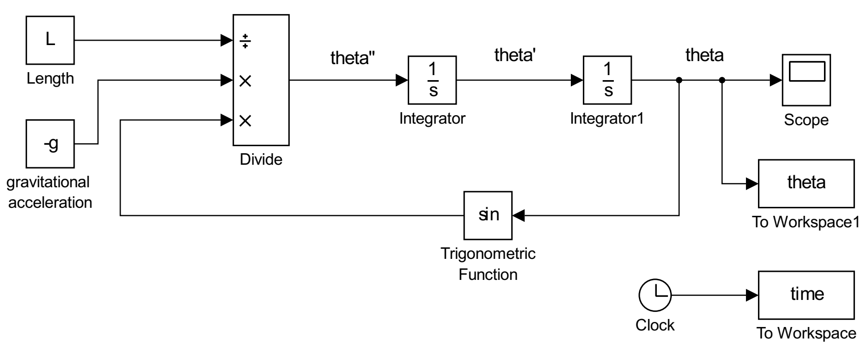

Figure 3.29: Nonlinear pendulum model with **To Workspace** blocks added to output $\theta(t)$ and t.

To see a plot of the solution, add the following lines to **pendulum.m**:

```
figure(1)
plot(time,theta)
xlabel('t')
ylabel('\theta')
```

Running the new **pendulum.m** m-file produces the plot in Figure 3.30.

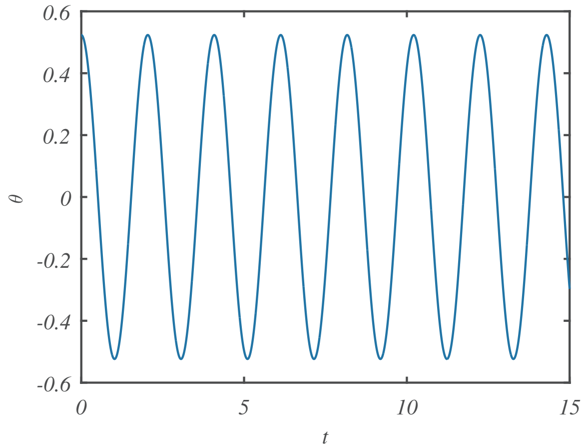

Figure 3.30: Plot of solution, $\theta(t)$ vs t, to the nonlinear pendulum model.

One can also animate the motion of the pendulum mass on the string. We use the data produces from Simulink to locate the position of the mass (as a ball) and the end of the string. For each time the mass and string are redrawn as we loop through time. The code to be added to **pendulum.m** is given as

```
rball=.05;             % mass radius
x=L*sin(theta);
y=-L*cos(theta);

    posx=x(1); posy=y(1);     % Mass's initial position

%Initialize figure, mass, and string
fig=figure(2);
axs=axes('Parent',fig);
ball=rectangle('Position',[posx-rball,posy-rball,2*rball,2*rball],...
               'Curvature',[1,1],...
               'FaceColor','b',...
               'Parent',axs);
rod=line([0 posx],[0 posy],'Marker','.','LineStyle','-')
axis(axs,[-L,L,-L-rball,L]);

for j=2:length(time)
    set(ball,'Position',[x(j)-rball,y(j)-rball,2*rball,2*rball]);
    set(rod,'XData',[0 x(j)],'YData',[0 y(j)]);
    axis([-L,L,-L-rball,L])
    pause(0.1);
end
```

In Figure 3.31 we show the starting location of the pendulum simulation.

It is interesting to compare the linear and nonlinear pendulum solutions

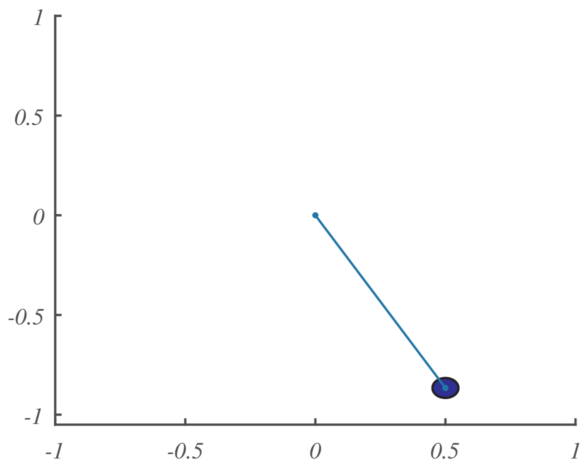

Figure 3.31: Simulation of the nonlinear pendulum in MATLAB.

to see when the small angle approximation holds. This can be done by combining the two models and comparing the solutions in a scope plot. Figure 3.32 shows such a model. The equations being solved are

$$\begin{aligned} \theta'' + 4\theta &= 0 \\ theta'' + 4\sin\theta &= 0. \end{aligned} \tag{3.9}$$

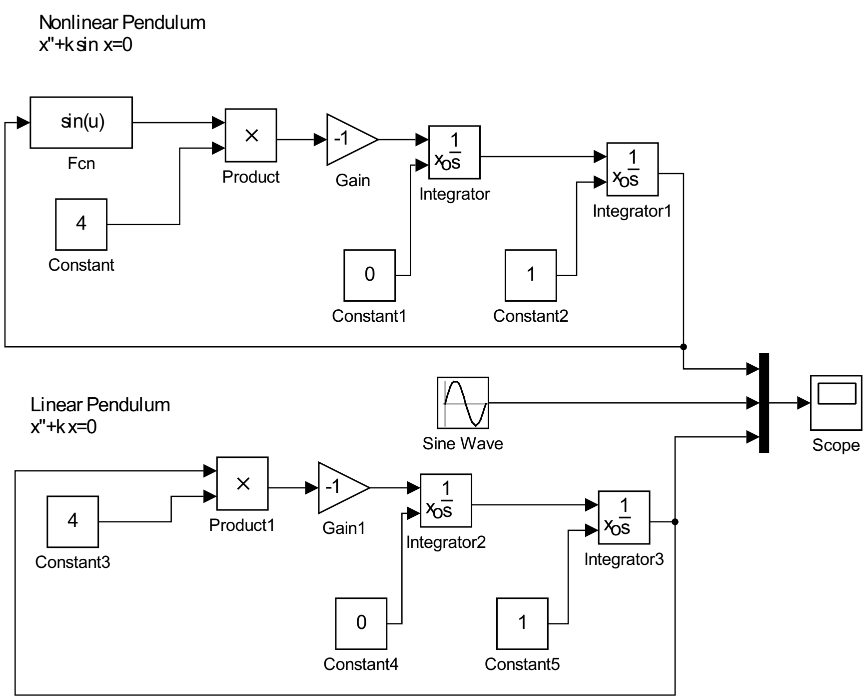

Figure 3.32: Linear and nonlinear pendulum.

3.5 Second Order ODEs in MATLAB

WE CAN ALSO USE **ode45** TO SOLVE second and higher order differential equations. The key is to rewrite the single differential equation as a system of first order equations. Consider the simple harmonic oscillator equation, $\ddot{x} + \omega^2 x = 0$. Defining $y_1 = x$ and $y_2 = \dot{x}$, and noting that

$$\ddot{x} + \omega^2 x = \dot{y}_2 + \omega^2 y_1,$$

we have

$$\begin{aligned} \dot{y}_1 &= y_2, \\ \dot{y}_2 &= -\omega^2 y_1. \end{aligned}$$

Furthermore, we can view this system in the form $\dot{\mathbf{y}} = \mathbf{y}$. In particular, we have

$$\frac{d}{dt}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} y_1 \\ -\omega^2 y_2 \end{bmatrix}$$

Now, we can use **ode45**. We modify the code slightly from Chapter 1.

```
[t y]=ode45('func',[0 5],[1 0]);
```

Here [0 5] gives the time interval and [1 0] gives the initial conditions

$$y_1(0) = x(0) = 1, \quad y_2(0) = \dot{x}(0) = 0.$$

The function **func** is a set of commands saved to the file **func.m** for computing the righthand side of the system of differential equations. For the simple harmonic oscillator, we enter the function as

```
function dy=func(t,y)
omega=1.0;
dy(1,1) = y(2);
dy(2,1) = -omega^2*y(1);
```

There are a variety of ways to introduce the parameter ω. Here we simply defined it within the function. Furthermore, the output **dy** should be a column vector.

After running the solver, we then need to display the solution. The output should be a column vector with the position as the first element and the velocity as the second element. So, in order to plot the solution as a function of time, we can plot the first column of the solution, y(:,1), vs t:

```
plot(t,y(:,1))
xlabel('t'),ylabel('y')
title('y(t) vs t')
```

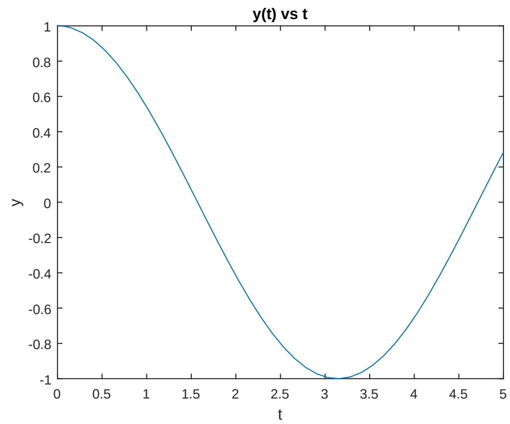

Figure 3.33: Solution plot for the simple harmonic oscillator.

The resulting solution is shown in Figure 3.33.

We can also do a phase plot of velocity vs position. In this case, one can plot the second column, y(:,2), vs the first column, y(:,1):

```
plot(y(:,1),y(:,2))
xlabel('y'),ylabel('v')
title('v(t) vs y(t)')
```

The resulting solution is shown in Figure 3.34.

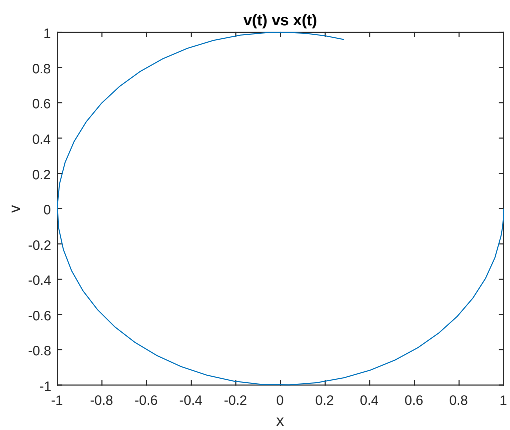

Figure 3.34: Phase plot for the simple harmonic oscillator.

Finally, we can plot a direction field using a quiver plot and add solution curves using **ode45**. The direction field is given for $\omega = 1$ by dx=y and dy=-x.

```
clear
[x,y]=meshgrid(-2:.2:2,-2:.2:2);
dx=y;
dy=-x;
quiver(x,y,dx,dy)
axis([-2,2,-2,2])
xlabel('x')
ylabel('y')
```

```
hold on
[t y]=ode45('func',[0 6.28],[1 0]);
plot(y(:,1),y(:,2))
hold off
```

The resulting plot is given in Figure 3.35.

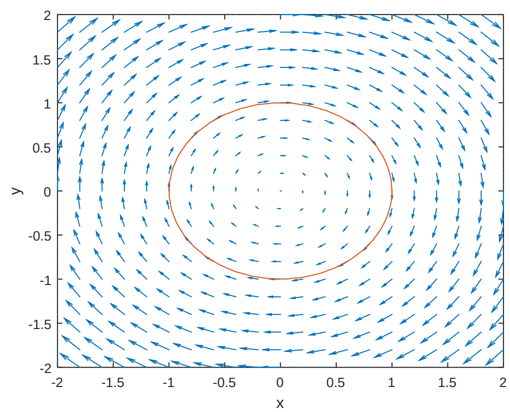

Figure 3.35: Phase plot for the simple harmonic oscillator.

3.6 Exercises

1. Model the following initial value problems in Simulink and compare solutions to those using ode45.

   a. $y'' - 9y' + 20y = 0,, \quad y(0) = 0, \quad y'(0) = 1.$
   b. $y'' - 3y' + 4y = 0, \quad y(0) = 0, \quad y'(0) = 1.$
   c. $8y'' + 4y' + y = 0, \quad y(0) = 1, \quad y'(0) = 0.$
   d. $x'' - x' - 6x = 0$ for $x = x(t), \quad x(0) = 0, \quad x'(0) = 1.$

2. Model the given equation in Simulink for an appropriate initial condition and plot the solution. Analytically determine and plot the solution and compare to the model solution.

   a. $y'' - 3y' + 2y = 10.$
   b. $y'' + 2y' + y = 5 + 10\sin 2x.$
   c. $y'' - 5y' + 6y = 3e^x.$
   d. $y'' + 5y' - 6y = 3e^x.$
   e. $y'' + y = \sec^3 x.$
   f. $y'' + y' = 3x^2.$
   g. $y'' - y = e^x + 1.$

3. Consider the model in Figure 3.36. Fill in the question marks with the correct expression at that point in the computation. What differential equation is solved by this simulation? [Put the equation in the simplest, recognizable form.]

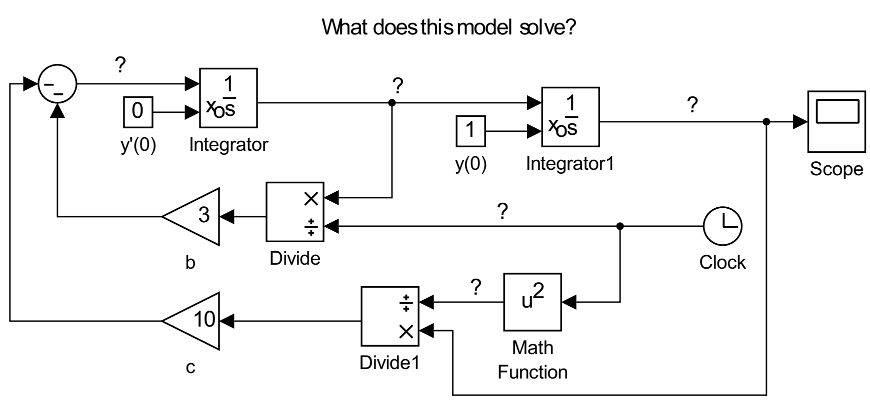

Figure 3.36: Mystery model for Problem 4.

4. Model the given equation in Simulink for an appropriate initial condition and plot the solution. Analytically determine and plot the solution and compare to the model solution.

   a. $x^2y'' + 3xy' + 2y = 0.$
   b. $x^2y'' - 3xy' + 3y = 0, \quad y(1) = 1, y'(1) = 0.$
   c. $x^2y'' + 5xy' + 4y = 0.$
   d. $x^2y'' - 2xy' + 3y = 0, \quad y(1) = 3, y'(1) = 0.$

e. $x^2y'' + 3xy' - 3y = x^2$.
f. $x^2y'' + 3xy' - 3y = x^2$.
g. $2x^2y'' + 5xy' + y = x^2 + x$.
h. $x^2y'' + 5xy' + 4y = 0$.
i. $x^2y'' - 2xy' + 3y = 0$.

5. Consider an LRC circuit with $L = 1.00$ H, $R = 1.00 \times 10^2\ \Omega$, $C = 1.00 \times 10^{-4}$ f, and $V = 1.00 \times 10^3$ V. Suppose that no charge is present and no current is flowing at time $t = 0$ when a battery of voltage V is inserted. Use a Simulink model to find the current and the charge on the capacitor as functions of time. Describe how the system behaves over time.

6. A certain model of the motion light plastic ball tossed into the air is given by
$$mx'' + cx' + mg = 0, \quad x(0) = 0, \quad x'(0) = v_0.$$
Here m is the mass of the ball, g=9.8 m/s^2 is the acceleration due to gravity and c is a measure of the damping. Since there is no x term, we can write this as a first order equation for the velocity $v(t) = x'(t)$:
$$mv' + cv + mg = 0.$$

a. Model this problem using Simulink.
b. Determine how long it takes for the ball to reach it's maximum height?
c. Assume that $c/m = 5\ \text{s}^{-1}$. For $v_0 = 5, 10, 15, 20$ m/s, plot the solution, $x(t)$, versus the time.
d. From your plots and the expression in part b., determine the rise time. Do these answers agree?
e. What can you say about the time it takes for the ball to fall as compared to the rise time?