

PHYS 305 Assignment 7: Numerical Solutions to Differential Equations via Euler and Runge-Kutta Methods.

Zachary Martin¹

¹University of Hawaii at Manoa

(Dated: 13 March 2019)

We employ the Euler and Runge-Kutta numerical methods to solve equations of motion for both a system of known analytic solution (a mass on a spring) and a system of unknown analytic solution (large angle pendulum). Using conservation of energy as a measure of accuracy, we find that fourth-order Runge-Kutta methods using step sizes of $dt = 1$ ms give fractional energy differences below $3 * 10^{-15}\%$ per cycle. We also find that unlike what small angle approximation solutions predict, the period of the pendulum does have a relationship to the initial angle amplitude, and it goes just as we would expect.

INTRODUCTION AND OVERVIEW

An important aspect of physics problems involves solving differential equations. These usually come from the equations of motion of the system. We would like to examine different approximations to solve differential equations via computational methods. The methods involved will be Euler's method and second-order and fourth-order Runge-Kutta method.

Mass on a Spring

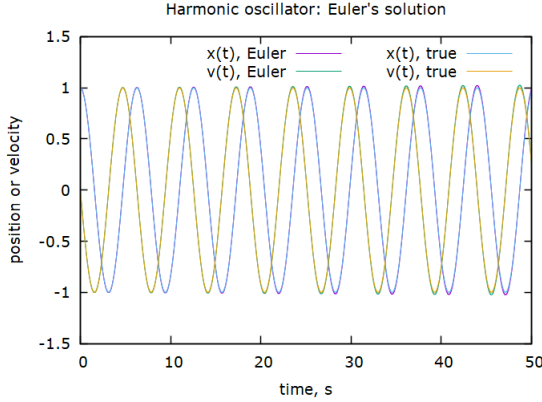


FIG. 1. Simple Harmonic Oscillator solved using Euler's method at step sizes of 1 ms. Note that the results are under the true lines. A fit to the numerical results gives the frequency $\omega = (0.999997 \pm 0.000005) \text{ rad s}^{-1}$ and a phase $\delta = (0.0009 \pm 0.0001) \text{ rad}$, which agrees extremely well with the true values of $\omega = 1 \text{ rad s}^{-1}$ and $\delta = 0$.

We start with a simple system for which we know the solution to: a mass on a spring. This system has

the equations of motion

$$\frac{dx}{dt} = v \quad (1)$$

$$\frac{dv}{dt} = -\frac{k}{m}x, \quad (2)$$

where k is the spring constant and m is the mass of the object on the spring. We know that the solution to this is simply

$$x(t) = x_0 \cos(\omega t) \quad (3)$$

$$v(t) = v_0 - \omega x_0 \sin(\omega t). \quad (4)$$

We can derive the energy at any time t for this system to be

$$E_s = \frac{1}{2}kx^2 + \frac{1}{2}mv^2. \quad (5)$$

Since energy is conserved, we can assume that the total energy is just the energy of the initial conditions at x_0, v_0 . Knowing this, we can experiment with different approximation techniques to calculate the solution to the differential equation and compare the results.

Large Amplitude Pendulum

Once we get a grasp of how the approximations work on a system we do know the exact solution to, we can solve another system that we don't know the exact solution to: a large amplitude pendulum. The general equations of motion are

$$\frac{d\theta}{dt} = \omega \quad (6)$$

$$\frac{d\omega}{dt} = -\frac{g}{L} \sin \theta, \quad (7)$$

where g is the gravitational acceleration and L is the length of the pendulum string. Note that these

equations are assuming the pendulum bob is held by an always tight string, or more so a massless rod, so that it falls always a distance L from the pivot point. We can derive that the energy at any time t during the swing will be

$$E_p = \frac{1}{2}m(\omega L)^2 + mgL(1 - \cos \theta) . \quad (8)$$

The differential equations have the same exact form as that of a mass on a spring if we apply the small angle approximation and let $\sin \theta = \theta$, for θ in radians. However, we are more interested in the general case where θ is chosen freely. Since our variable θ is inside the sine function, we cannot solve this exactly. It is necessary that we use numerical approximations.

THE COMPUTATIONAL PROBLEM

The importance of numerical approximations is evidence for the need of computational methods. We want to solve the equations of motions step by step through time, starting from our initial conditions and working our way through. One method of solving the coupled differential equations like 1 and 2 is called Euler's method. The idea of this method is to approximate each segment as a line, using the slope at each point to move to the next point. Thus, the approximation has the linear form [2]

$$y_{i+1}(t + dt) = y_i(t) + \left(\frac{dy}{dt} \right)_i (t) dt . \quad (9)$$

The method can work, but there are better ways to numerical approximate the solutions. The actual slope from point to another is changing of course, since the solutions are not linear. Euler's method does not account for this change in slope [1]. One way to account for this is to add another term that depends on the midpoint of the steps which will correct for the change in slope. This is the main idea behind the Runge-Kutta method. Since we are only making approximations, there are infinite orders or corrections to make in order to gain an exact solution. We are only interested in second-order (RK2) and fourth-order (RK4) solutions. The RK2 method uses

$$y(d + dt) = y(t) + k_2 \quad (10)$$

$$k_2 = dt f(t + \frac{dt}{2}, y(t) + \frac{k_1}{2}) \quad (11)$$

$$k_1 = dt f(t, y(t)) , \quad (12)$$

where we defined $f(t, y(t)) = \frac{dy}{dt}$. Notice that the correction terms are built upon the ideas from Euler's method. If we want to go to the fourth order

correction, we instead use the equations

$$y(t + dt) = y(t) + k_5 \quad (13)$$

$$k_5 = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (14)$$

$$k_1 = dt f(t, y) \quad (15)$$

$$k_2 = dt f(t + \frac{dt}{2}, y(t) + \frac{k_1}{2}) \quad (16)$$

$$k_3 = dt f(t + \frac{dt}{2}, y(t) + \frac{k_2}{2}) \quad (17)$$

$$k_4 = dt f(t + dt, y(t) + k_3) . \quad (18)$$

The equations of motion give us coupled differential equations. Thus, each method will have to be used on each coupled equation and they will build each other up from there.

At each step, we will want to measure how the energy is conserved. If the approximations are correct, we should expect to have constant energy. To measure the fractional difference in the energy per period, we will measure

$$\frac{(dE/E)}{T} = \frac{|E_t - E|/E}{T} . \quad (19)$$

Here, E_t represents the energy at each step, or each time t , and E is the total energy from the initial values. Conservation of energy per period will be our measure of how accurate the approximations are.

Clearly, this is a tedious task. A computer can calculate these approximations for a given step size dt in a short time.

Programming (Mass on a Spring)

We will need to do an iterative loop over time that will change the values of $x(t)$ and $v(t)$ and recycle the variables by setting them equal to the previous values and putting them back into the loop to get even higher values. For Euler's method, this is straight forward. We can write for the loop,

```
for (t=t0; t<Tmax; t+= dt){
    vt = vtold + dt*(-k/m)*xtold;
    xt = xtold + dt*vtold;
    xtold = xt; // old -> new
    vtold = vt;
    xtrue = xt0*cos(w*t);
    vtrue = vt0 - w*sin(w*t);
    Et = 0.5*k*xt*xt + 0.5*m*vt*vt;
    dE = fabs(Et - Etrue)/Etrue;
    // outfile << ... << endl;
}
```

For RK2 and RK4 methods, it is slightly more complicated. To make things simple, we will define our correction function in a separate file and call to it

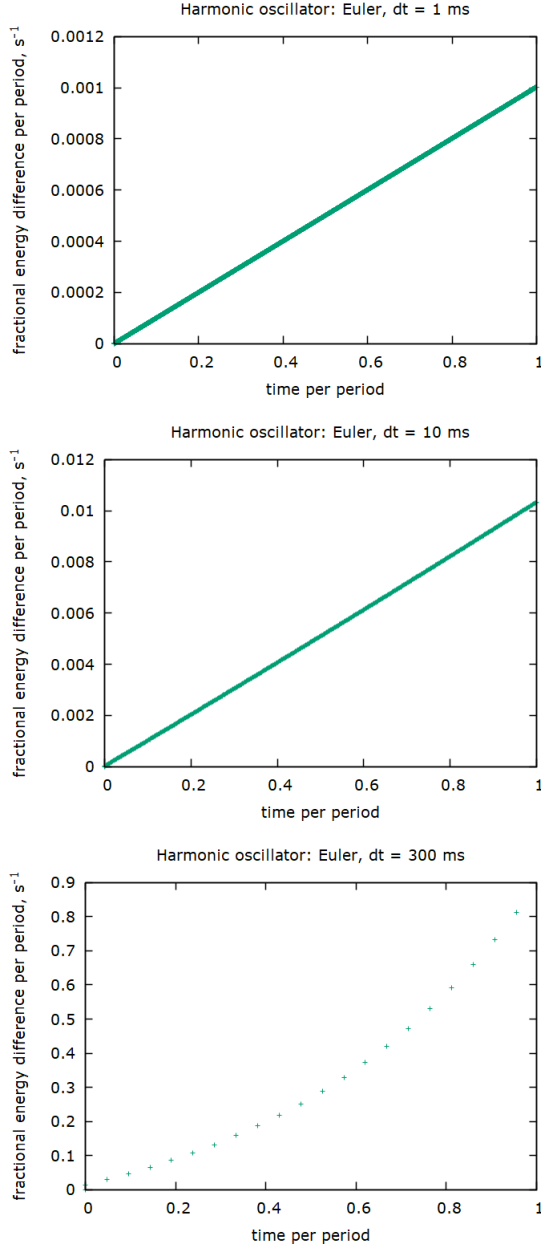


FIG. 2. Fractional differences of the calculated energies from the results of Euler's method. At low enough step sizes, the difference is fairly linear through time. We also see that the differences increase by a factor of 10 and then 100 going from $dt = 1$ mm to 10 mm and $dt = 10$ mm to 300 mm.

in our numerical evaluation. For instance, we can create a file for the RK2 correction which holds the function [4]

```
double FRK2xv(int ytype, double (*f_x)
(double, double, double), double (*f_v)
(double, double, double), double t,
```

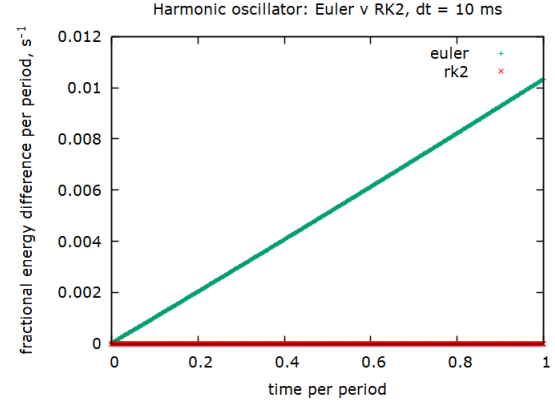


FIG. 3. A comparison of fractional differences in calculated energies from the results of Euler's method and second-order Runge-Kutta method. It is evident that the RK2 method is better than Euler's method in this numerical evaluation, as the RK2 differences look negligible compared to the Euler differences.

```
double xold, double vold, double dt)
{
    double k1x, k1v, k2;
    k1x = dt*(*f_x)(t, xold, vold);
    k1v = dt*(*f_v)(t, xold, vold);
    if(ytype==0){
        k2 = dt*(*f_x)( t, xold+
            k1x/2.0, vold+k1v/2.0 );
    }else{
        /* ytype = 1 */
        k2 = dt*(*f_v)( t, xold+
            k1x/2.0, vold+k1v/2.0 );
    }
    return(k2);
}
```

The RK4 method will be exactly the same, but with the higher order terms. Then we can simply implement these into the program for the numerical evaluation as

```
for(t=t0; t<Tmax; t+= dt){
    vt = vtold +
    FRK2xv(1,f_x,f_v,t,xtold,vtold,dt);
    xt = xtold +
    FRK2xv(0,f_x,f_v,t,xtold,vtold,dt);
    xtold = xt; // old -> new
    vtold = vt;
    ...
}
```

where we have defined the functions f_x and f_v to be the equations of motion (the subscript indicates which variable is being differentiated).

We are now ready to analyze the results.

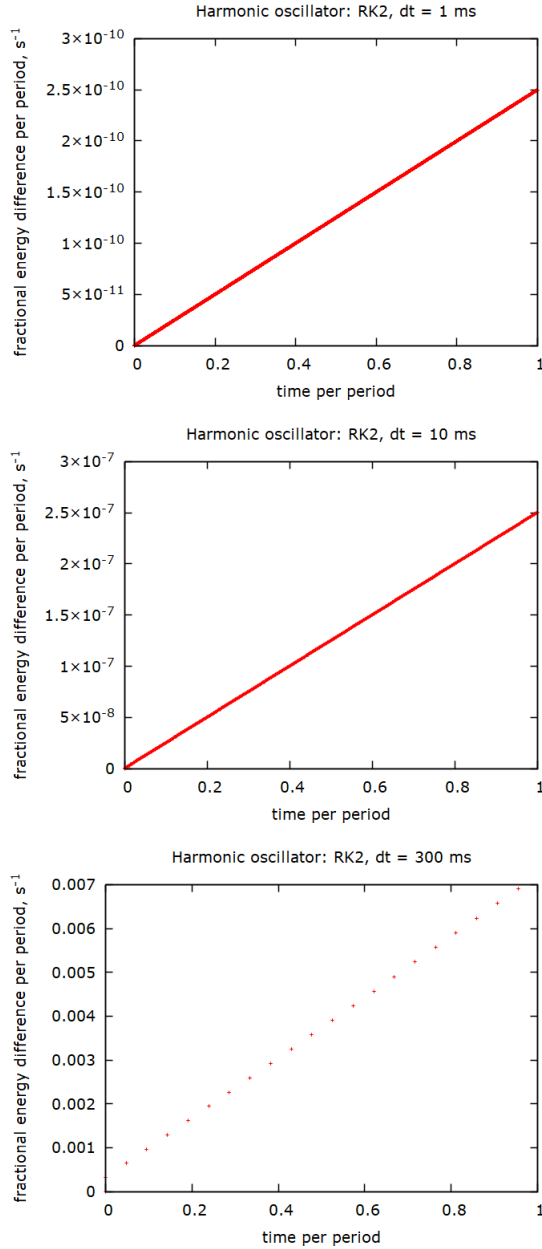


FIG. 4. Similar to Figure 2. We can still see a linear behavior, with the last plot more linear than the last plot for the Euler method results. Also, the change in fractional difference for varying step sizes is more significant for the RK2 method.

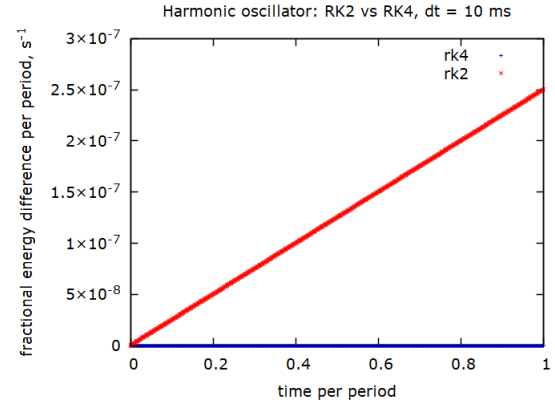


FIG. 5. A comparison of the RK2 and RK4 methods. Like the comparison between Euler and RK2, the RK4 results are negligible relative to the RK2 results, and thus the RK4 method is better.

RESULTS AND GRAPHS

DISCUSSION AND ANALYSIS

Mass on a Spring

We could numerically approximate the mass on a spring system using Euler's, RK2, and RK4 methods and Figures 2 - 6 show us how they all compare. Immediately, we can see that Euler's method gives agreeing results. Figure 1 shows just how close the approximation gets to the true solution just at step sizes of 1 ms. The fitted values for the frequency and phase of the system are well within 1σ of the true values. Taking a look at the energy conservation, we see from Figure 2 that energy is well conserved at 1 ms and 10 ms. It is conserved within a fractional difference of 1% per cycle at a maximum step size of about 9 ms. For such a simple method of numerical evaluation, this is impressive. But we can also see how larger step sizes break the evaluation. In 300 ms step sizes, the energy is almost at a 100% fractional difference after one cycle. This is clearly not good, and it is something we must avoid when performing simulations. Some systems may require extremely small step sizes. Or instead, we can try another method. From Figure 3, we can see that the fractional difference in energy from the RK2 results is nothing relative to the Euler results. Figure 4 reveals that the differences are about one million times smaller for step sizes of 1 ms. This is a clear reason why the Runge-Kutta method is favorable over Euler's method. However, we can also see that increas-

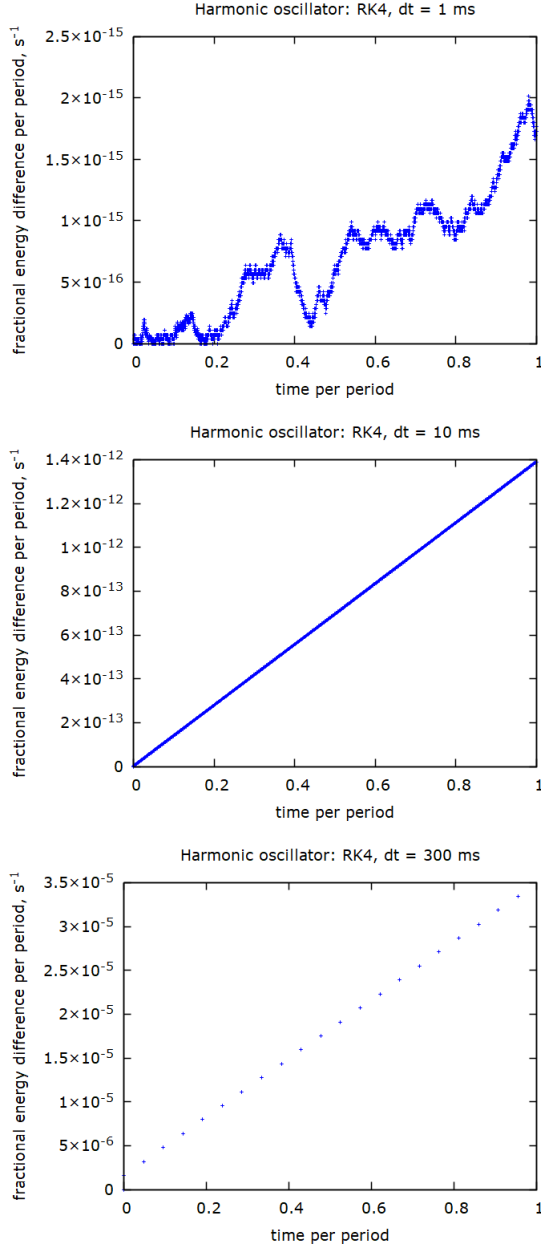


FIG. 6. Similar to Figures 2 and 4. We get the trend of linear fractional difference for the last two plots, but not for when $dt = 1$ ms. We can see that it is on the order of 10^{-15} , far less than the results from the other methods, and the differences across varying step sizes are even more significant.

ing the step sizes in the same way reveals significant changes in the fractional differences. But even so, at a step size of 300 ms, the fractional difference after one cycle is still less than 1%. This is improved even further using the RK4 method. Figure 5 shows that

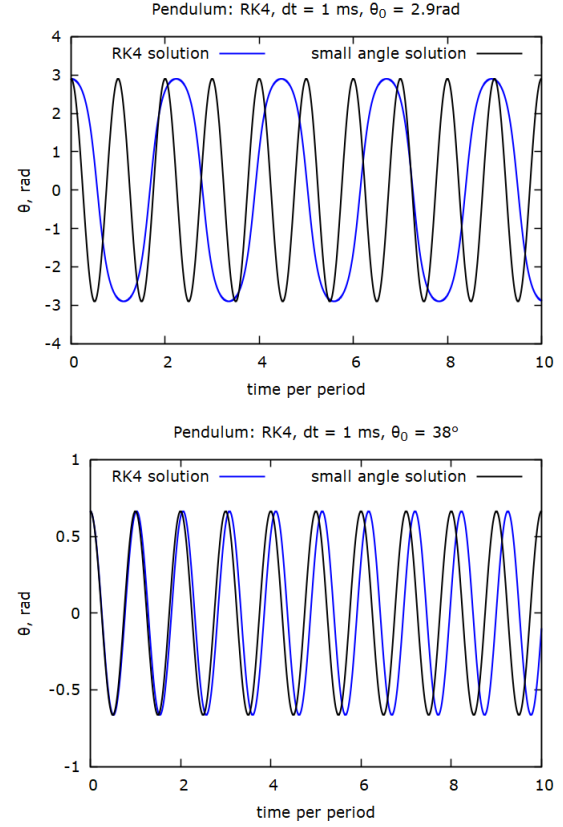


FIG. 7. Comparing the RK4 solution to the 'true' solution using small angle approximation. It is clear that the small angle approximation is very different from the real solution. We can see that the numerical evaluation at smaller angles does begin to converge to an agreement with the small angle approximation solution, but the approximation breaks over time, as we would expect.

relative to the RK2 results, the RK4 results are negligible. And again, from Figure 6, we can see that it yields fractional differences 100000 times smaller at step sizes of 1 ms. It may show an even larger jump of 10^8 from 10 ms to 300 ms step sizes, but the fractional difference in energy after one cycle at that large step size is very well below 1%, much better than the RK2 result. We do notice some change in behavior from the RK4 results. We could see some pattern of linear behavior from Figures 2, 4, and even the last two of Figure 6. But the first plot in the RK4 figure fluctuates and is not at all linear. It may be certain 'glitches' pertaining to the calculation or even the capabilities of the computer. The numbers there are so small that precision can easily be lost. We had found before how machine precision can affect the values of our calculations [3]. It may also be a break in the calculation. Certain values

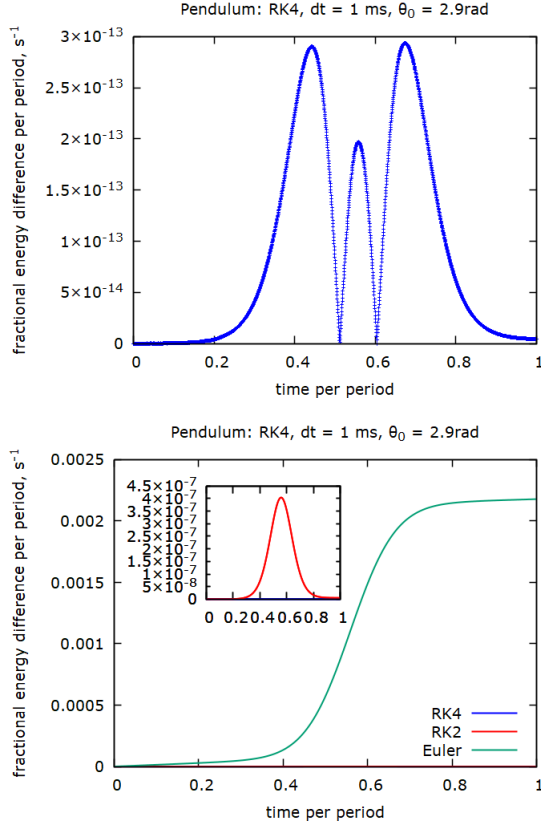


FIG. 8. On top, we can see the fractional energy differences for the pendulum over one period using RK4. We can also see how it compares to the results of the other methods (Euler, RK2) below that. We can see that there is no longer a linear behavior, but more so recognizable and non-fluctuating behavior. The comparison also shows how far the results are from each other, each making the previous negligible.

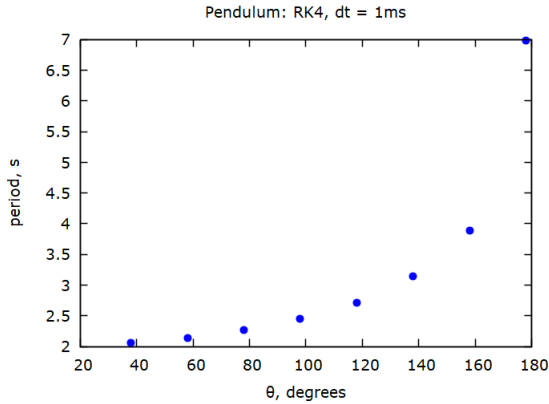


FIG. 9. The period of the $\theta(t)$ functions obtained from the RK4 method for varying starting amplitudes/angle.

can lead to a certain "error resonance" in recursive approximations like Euler and RK. In any approximation, there are always many values that just do not work for the formula. Since 10 ms and 300 ms showed the expected linear behavior, it may be that 1 ms may be a certain value that breaks RK4. But of course, the fluctuations are so small that actually, we can still use RK4 at such a step size and still achieve accurate results. It is therefore evident that we should use RK4 for the unknown problem, the large amplitude pendulum.

Large Amplitude Pendulum

We can first confirm our results for the known solution at small angles (small angle approximation). From Figure 7, the data curve converges towards the true curve as we go to smaller angles. Even at 38° , the approximation agrees fairly well for the first few cycles, revealing an upper limit for the reliability of the small angle approximation. At larger angles, we can see that the period of the angle oscillation is around two times greater than what the small angle approximation predicts. When we look at the fractional energy differences through time, we see that a step size of 1 ms gives a non linear behavior in all of the approximation methods. This may show that the 'outlier' behavior in Figure 6 may not be an error in calculation, but more so an underlying behavior of the actual function. The behaviors in Figure 8 are not chaotic. I do not know what is the underlying cause of this. It may be due to the functions of $\theta(t)$ and $\omega(t)$ in the equation for energy creating a superposition of sinusoidal waves. Whatever the case, it is clear how small the fractional differences are compared to each method, with the Euler method showing the greatest difference, but still much less than 1%. If we take a look at how the period of the pendulum varies as a function of the angle in Figure 9, we can see that there is definitely a relationship between the angle and the period. This is not what we would expect if we used small angle approximation for all angles. We would instead predict that there is no relationship. But from the figure, we can confirm that at smaller angles, which we can quantify is around 40° by reading off the curve, there is a constant behavior. Recall that these results are assuming a model of a massless rod holding the pendulum bob, otherwise at such large angles, the mass would fall straight down with a behavior not described by the equations of motions we used. Then it makes sense that the period behaves like we see in the figure. As the angle approaches 0, the period

should also approach 0. And as the angle gets closer to 180° , the period should get longer and longer, as gravity will fight the normal force of the rod, resulting in slower accelerations in the beginning, which approaches infinity (i.e. the rod stands straight up at 180° forever).

CONCLUSIONS

The Euler and Runge-Kutta methods have proven to both be useful tools in solving differential equations numerically. We found that the RK method worked better, but of course, it relies on the techniques employed by the Euler method. Applying both of these to a system we knew the solution to revealed the behaviors that may come up when doing a numerical evaluation over an analytic one. We are going to get error inevitably by using approximations, we can find how changing our step size can reduce that error. And once we understood how the numerical methods applied to the known system, we could apply it to a system with an unknown analytic solution and discover some behaviors that we

may have otherwise not predicted. This has proven to show the capabilities of computational methods. With fairly simple mathematical techniques, we can discover the behaviors of multiple types of systems so long as we can first find the equations of motion.

ACKNOWLEDGMENTS

-
- [1] Gorham, P., "Numerical solutions of differential equations: Runge-kutta," <https://www.phys.hawaii.edu/~gorham/p305/DiffEq2.html> (2019), accessed on 2019-03-13.
 - [2] Landau, R. H., *Computational Physics: Problem Solving with Computers* (WILEY-VCH Verlag GmbH & Co. KGaA. Weinheim, 2007).
 - [3] Martin, Z., "Phys 305 assignment 2: Analyzing the behavior of a bouncing ball with energy loss per bounce and machine precision." (2019).
 - [4] Weisstein, E. W., "Runge-kutta method," <http://mathworld.wolfram.com/Runge-KuttaMethod.html>, from MathWorld—A Wolfram Web Resource. Accessed on 2019-03-13.