Finding Optimal One Step Method For Duffing Oscillator

Moises Sanchez Perm: 9866922

February 23, 2019

Abstract

1 Introduction

The harmonic oscillators are some of the important ODEs because of their application in modern engineering and physics. However, what is important to know is that this is just a special case of an equation called the Duffing Oscillator. The Duffing Oscillator is a second order ODE, with constants $\beta, \delta, \omega, \gamma,$ and α.

$$\ddot{x} + \delta\dot{x} + \alpha x + \beta x^3 = \gamma\cos(\omega t) \tag{1}$$

These constants represent the nonlinearity, dampening, angular frequency, amplitude, and linear stiffness respectively. Changing these parameters represents various types of oscillators from dampened, driven, chaotic, and etc. With the application of the our group wanted to find the best one step method to solve the equation. In order to accomplish this we compared Euler's Method, Modified Euler's method and Runge Kutta because they are simple to code, widely accepted in the field of simple ODE computation, and offer accurate enough results. We then varied the step and iterations to check how the numeric methods responded to small and large steps and iterations. We also checked how each system conserved energy by computing the energy variance and comparing which method had the smallest energy variance. The smallest energy variance meant the method had less error. Finally we checked at phase diagram at the presences of chaotic attractors to see which method did best at representing chaotic behavior.

2 Methods

However, the issue we run into is that we have solving for a second order nonlinear ODE. When solving for the Duffing Oscillator, we can not simply just plug it into the general numerical methods in the book, so we had to first had to find a way to code a second order ODE. The method we decided to do was to first use change of variables and rewrite the second order ODE as a system of first order ODEs.

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} y \\ -\delta\dot{x} - \beta x - \alpha x^3 + \gamma\cos(\omega t) \end{pmatrix} \tag{2}$$

Remark. When we translate this system of ODE to code for numerical methods, we transform x, and y into w_2 and w_1 respectively.

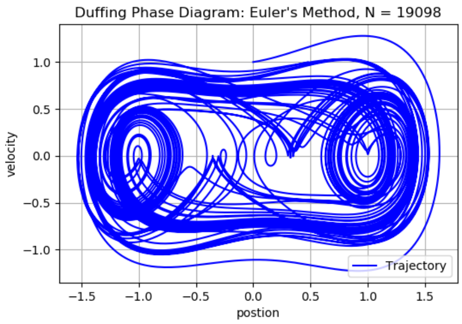
After rewriting the equation as the system ODEs we then simply just solved for the system of equation using Euler, Modified Euler, and Runge Kutta. We ran this numerical methods using fixed time steps of 0.05, and 0.005 as well as fixed iterations of 1000, and 10,000. Finally, we then just found the computation time, phase graph, energy variances, and table that contains velocity and position.

3 Validating Method

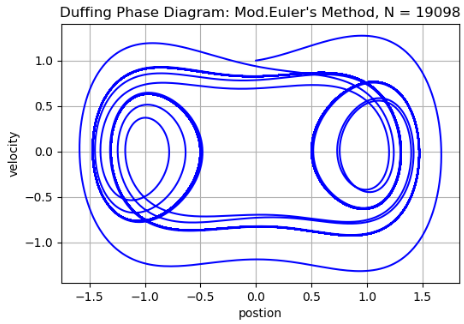
When checking whether we could use these numerical methods to solve the Duffing Oscillator we had to first check if our equation actually had any solutions. We checked for the existence of solutions by checking either our ODE was well-posed. Furthermore, by checking for Lipschitz continuity, we saw that our equation was Lipschitz continuous and therefore had a solution by Theorems 5.4 and 5.6 in Burden[2]. Afterwards we translated our numerical methods to work for systems of equations. We then fixed our time steps, and iterations. The reason for this is because we needed to see how each iteration responded to certain parameters for machine errors. We then solved for the solutions and the variance of energy. Our group's reason for finding the energy variance was to check how much energy was being conserved. Since we did not have an actual model with exact solutions, calculating the energy variance was the best and most general way to see which method was more accurate. Then we checked the phase diagrams to observe the behavior at the appearance of chaotic attractors. Our group did not check the values of the tables in this case was because it near impossible to check thousands of iterations of numbers to see which method look the most chaotic. Rather than check many values, we simply just check the phase as it would really easy to see which look chaotic than the rest. Finally we checked the time, because it is important to have a method that outputs a solution in a reasonable amount of time.

4 Results

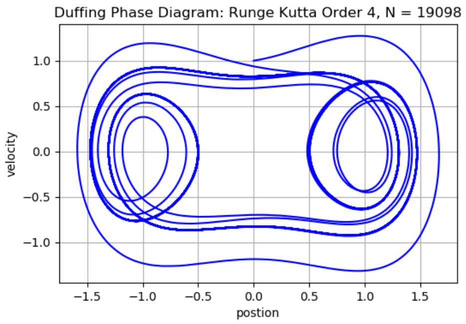
After running our code the first thing we saw that Euler's method ran two times fast than that of Runge Kutta, and Modified Euler's method. Interestingly we found that the solutions for Runge Kutta, and Modified Euler's were very similar, with Euler's differing from them by three decimal places. One of the more notable things to occur would be Euler's method had the smallest energy variance for all test, but except when $\gamma = \delta = 0$. This meant that Runge Kutta, and Modified Euler are best when our system is undamped and driven. This idea is further supported when observing the behavior of each method under of step and iteration changes. If our step size was too small, for e.g step < 0.05 and iteration N > 10000, then we encounter an error of overflow for both Kunge Kutta and Modified Euler's method.



(a) 1a: Euler Method



(b) 1b: Modified Euler Method



(c) 1c: Runge Kutta Method

Figure 1: Here we have the phase diagram, with the conditions $\delta = 0.3, \alpha = -1, \beta = 1, \gamma = 0.5$, and $\omega = 1.2$. In this situation the parameter gamma is what is creating the chaotic behavior.

Table 1: Duffing Positions: Runge Kutta, Modified Euler, and Euler At 0.05 Step

Time t:	Runge Kutta	Modified Euler	Euler
0.0	1.0	1.0	1.0
0.025	1.0049658094648881	1.004965936148431	1.005
0.05	1.0105337218873722	1.0105225272836336	1.0105846039081416
0.075	1.0166994606783055	1.0166542099111706	1.016739267881904
0.1	1.0233428938108888	1.0233432901912982	1.0234474434780407
0.125	1.0305696992166058	1.030569868699134	1.0306904923120614
0.15	1.0383685566499212	1.038311749981913	1.0384475854828041
0.175	1.0464318634559255	1.0465443373800738	1.0466955880418285
0.2	1.055039235735802	1.0552405136960767	1.0554089288821351
0.225	1.0641712131547016	1.064370508433189	1.064559456538068
...	...	...	...
477.225	0.8248916053802484	0.8255149575388966	0.9358141204675647
477.25	0.8249619412583896	0.8252769373009089	0.9425910138512372

Table 2: Duffing Positions: Runge Kutta, Modified Euler, and Euler At 0.25 Step

0.0	1.0	1.0	1.0
0.2	1.0437228182096867	1.0440679055703503	1.05
0.5	1.1442680624150914	1.1407796947475073	1.1517700884844508
0.8	1.2325967852014565	1.2283259395413146	1.2644706862360593
1.0	1.2235962077352698	1.2248949005403684	1.301295910895475
...	...	...	...
475.2	nan	nan	-0.873088339458341
475.5	nan	nan	-0.8901653595508654

Remark. This is a chart where we have a N = 100000 and a time step of 0.05. Take note that the numbers look very similar, but the behaviors on the charts are drastically different. Also on the other hand table 2, by changing step to 0.25 and lower iterations to N = 10000 we get overflow. So having correct step and iterations is important to having accurate numbers for Modified Euler and Runge Kutta.

This shows us that Runge Kutta and Modified Euler must have be a correct step and iteration in order to work. When checking out the phase diagrams, at the creation of chaotic attractors, Euler demonstrates the most accurate behavior. Also take notice that while Modified Euler look very similar to Runge Kutta, it contains more periodic doubling than it and is closer to reaching full chaotic behavior like Euler's Method. An explanation by Dr. Zidan, and Dr. Salma for the effectiveness for Euler's Method representation of chaos would be that the extra order of error gives an a nonlinear term which helps represents nonlinear behavior [3].

5 Conclusion

Overall, we found that Euler's method does the best when trying most cases of the Duffing Oscillator due to its extra degree of error. However, when our system is missing the term that creates chaos attractors is zeroed out then a Modified Euler's Method does the best. We were able to make this conclusion based on the energy variance, table behaviors at different parameters, and the phase diagrams. While Euler's method was successful for Duffing Oscillator it would be a good project to check if Euler's Method is successful for all general nonlinear ODE with the potential of producing chaotic attractors.

References

[1] Bjorn Birnir. *Dynamical Systems Theory*

[2] Richard L. Burden, J. Douglas Faires, and Annete M. Burden. *Numerical Analysis 10th Edition.* Cengage Learning.

[3] Zidan MA, Radwan AG, Salama KN (2011) The effect of numerical techniques on differential equation based chaotic generators. ICM 2011 Proceeding. doi:10.1109/ICM.2011.6177395