

Ph20 HW3

Nathan T Barton

February 8, 2019

1 Introduction

During this week, I wrote a python program to investigate numerical techniques for solving differential equations, as applied to the simple harmonic oscillator system. The code for the main program is hw3.ipynb, which defines functions for each of the numerical techniques implemented, and generates the corresponding figures.

The Euler methods were implemented using numpy arrays and matrices, so that each time-step could be computed as the product of the previous time-step and a time-step matrix. This made the code efficient and easy to modify to implement the various methods. In order to directly compare the methods, all methods were called with the same parameters, notably with $x_0 = 1$, $v_0 = 0$, a max time of $t_{max} = 25$, and time-step $h = 0.05$

2 Results

2.1 Explicit Euler Method

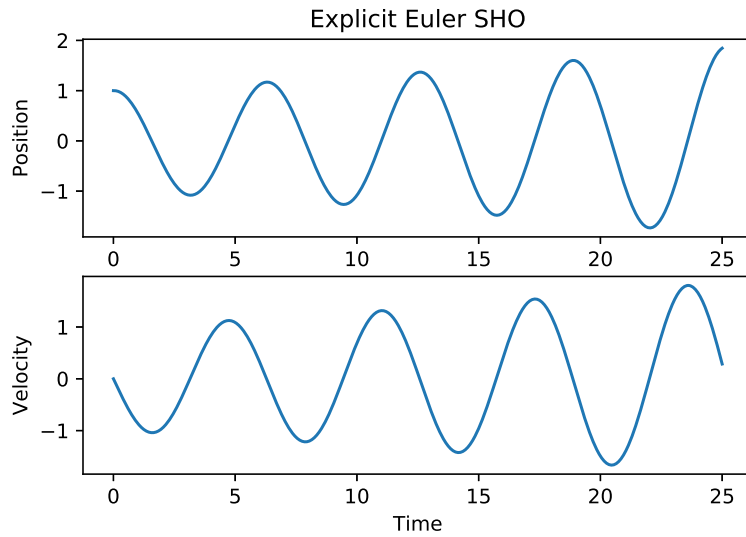


Figure 1: Oscillations of a SHO using the Explicit Euler Method

In Figure 1, we can see sinusoidal-like oscillations of position and velocity for the explicit Euler method. However, we can notice a slight increasing trend in amplitude, as the errors accumulate.

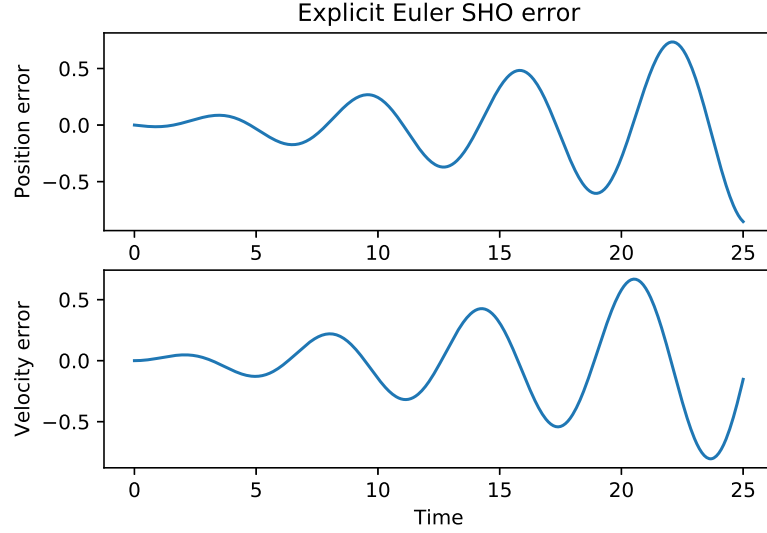


Figure 2: Error of SHO using Explicit Euler Method

To better show the trend in error, in Figure 2, the error in the position and velocity of the oscillator is plotted over time. The error was computed first by computing the analytic solution for the harmonic oscillator, and finding the difference between the expected analytic value and the numerically computed value. Here, we can see that the error reaches local maxima with each crest/trough of the oscillator waveform. The error directly out of phase with the oscillator, and steadily grows with time.

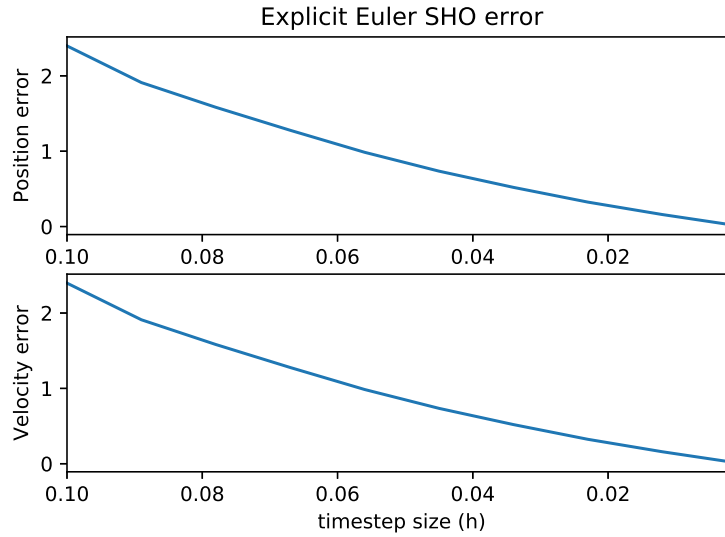


Figure 3: Maximum error of Explicit Euler SHO over 25 oscillations versus time-step size

Next, I compared the error of the Explicit Euler SHO for varying time-step values in order to see the truncation error. In Figure 3, we see that, for small values of h , the error decrease is roughly linearly proportional to the time-step size

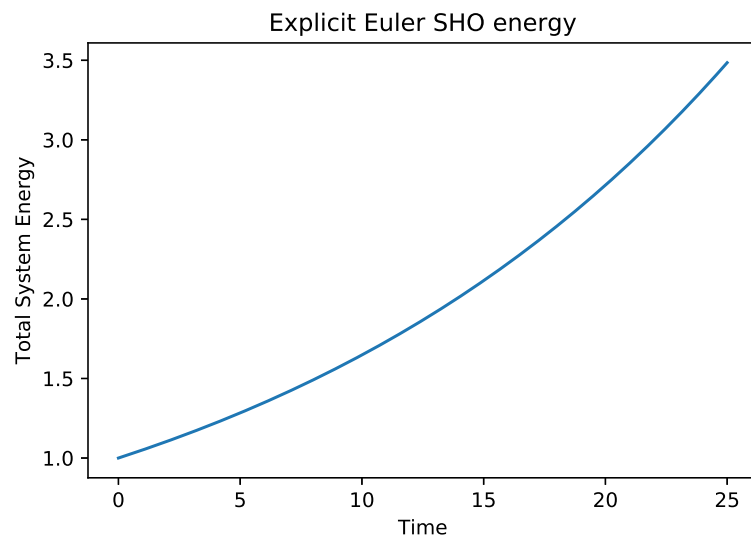


Figure 4: Energy of Explicit Euler SHO over time

In Figure 4, we see that the energy of the system, as modelled by the Explicit Euler method, increases roughly exponentially with time. This is expected from the increasing amplitude as seen in figure 1, and would limit the utility of the method for accurately modelling the amplitude of the system over long time scales.

2.2 Implicit Euler Method

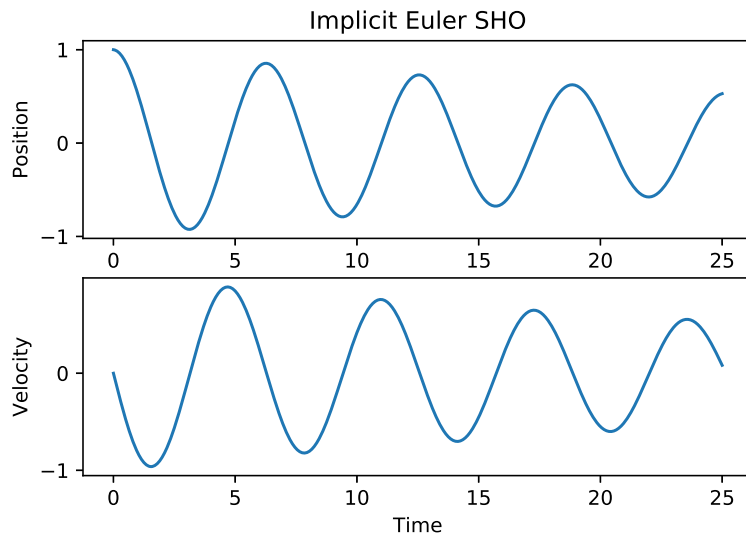


Figure 5: Oscillations of a SHO using the Implicit Euler Method

In Figure 5, as with the Explicit Euler method, we can see sinusoidal-like oscillations of position and velocity for the implicit Euler method. However, unlike with the Explicit method, we can notice a slight decreasing trend in amplitude.

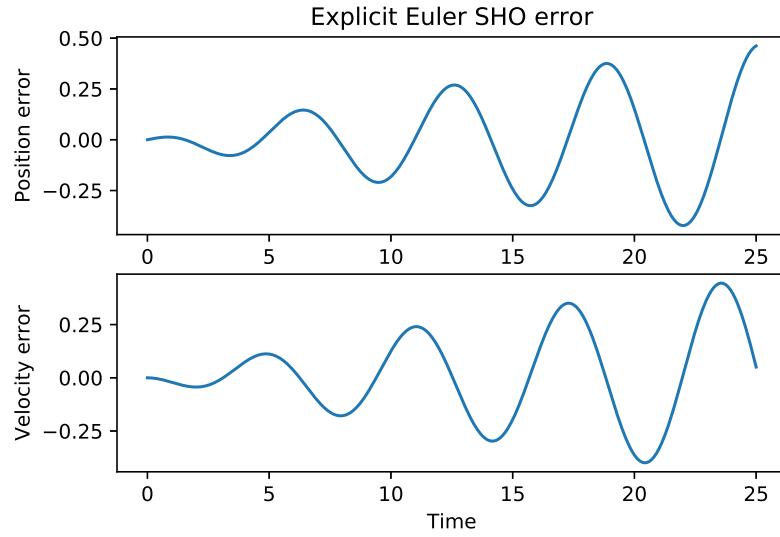


Figure 6: Error of SHO using Implicit Euler Method

Figure 6 shows the error in the position and velocity of the oscillator over time. Similar to the Explicit method, we can see that the error reaches local maxima with each crest/trough of the oscillator waveform. However, in this case, the error is in phase with the oscillator, and steadily grows with time, resulting in a decreasing amplitude over time.

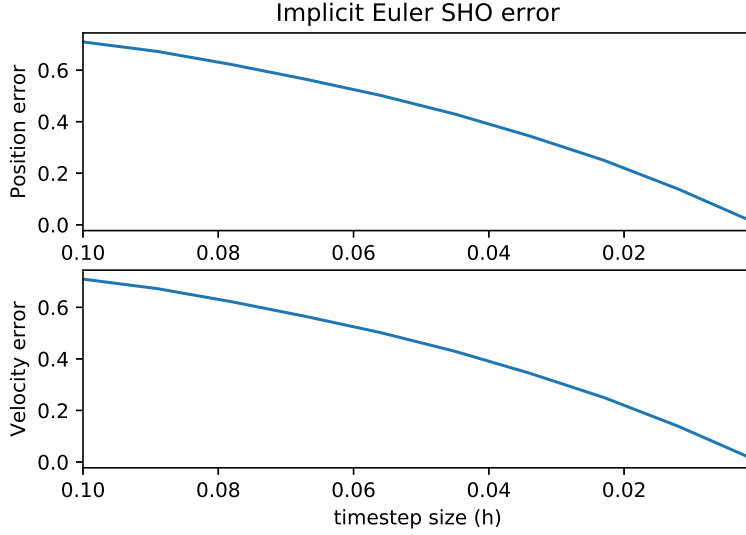


Figure 7: Maximum error of Implicit Euler SHO over 25 oscillations versus time-step size

Figure 3 shows the truncation error for the implicit Euler method. Once again, we see that, for small values of h , the error decrease is roughly linearly proportional to the time-step size. However, unlike the Explicit method, the truncation error is concave downwards with decreasing time-step size, and thus will scale better with increasing precision.

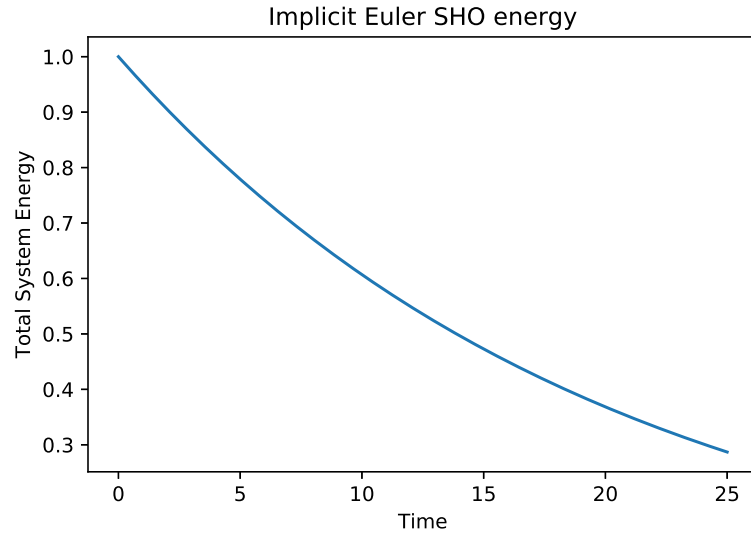


Figure 8: Energy of implicit Euler SHO over time

Figure 8 shows that the energy of the system, as modelled by the Implicit Euler method, decreases roughly exponentially with time. This is expected from the decreasing amplitude as seen in figure 5, and once again would limit the utility of the method for accurately modelling the amplitude of the system over long time scales.

2.3 Phase-space of Euler methods and Symplectic Method

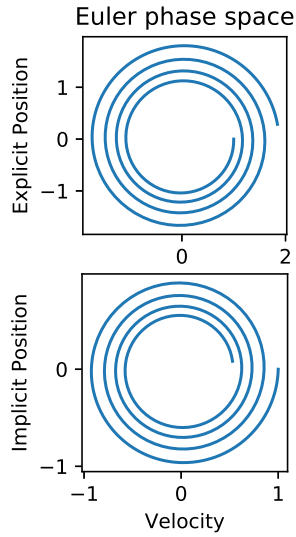


Figure 9: phase-space plot for Explicit and Implicit Euler SHO

Figure 9 shows the phase space plot for the Explicit and Implicit Euler methods. We can see that the Explicit method spirals outward, while the Implicit method spirals inward as a result of the increasing/decreasing energies of the modeled systems respectively.

In order to reduce this energy error, I implemented the Symplectic Euler method. In Figure 10, we can see that, indeed, the phase space plot produces a closed curve, and thus energy error is conserved in this model.

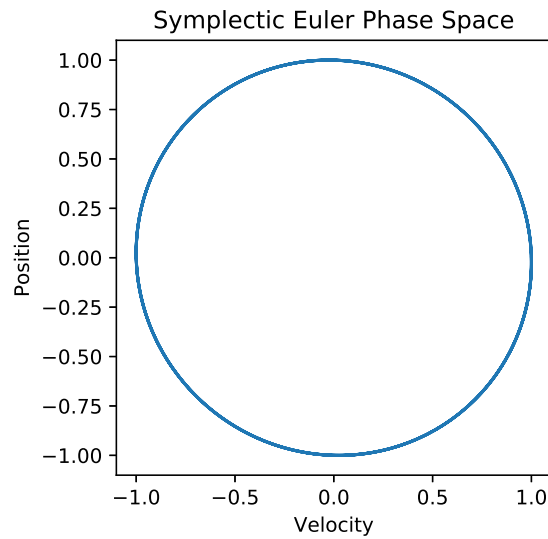


Figure 10: phase-space plot for Symplectic Euler SHO

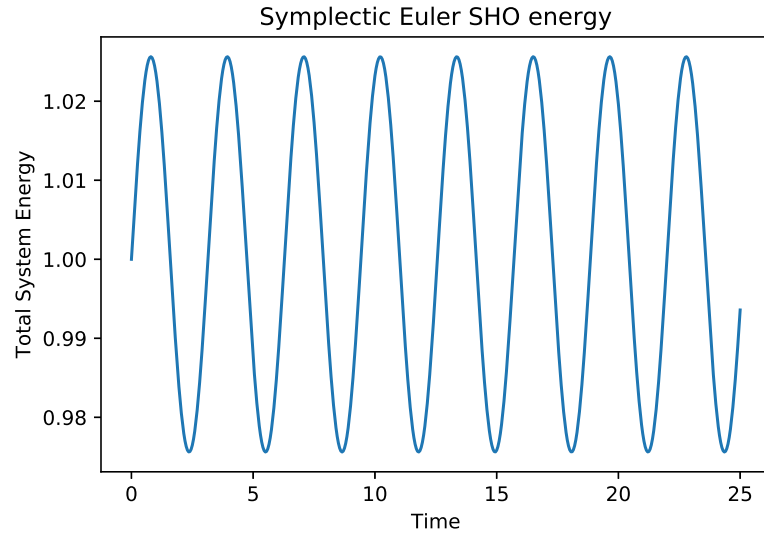


Figure 11: Energy plot for Symplectic Euler SHO

In Figure 11, we can see that the error in energy for the system oscillates with time, yet remains bounded. Therefore, the error in the amplitude, and thus energy of the system, will not grow over long timescales.

To study the other sources of error with the Symplectic method, in Figure 12, the position and velocity is plotted out to time $t_{max} = 10000$. Here, we can see that the energy/amplitude of the system is modelled correctly, however the Symplectic method introduces a phase shift error. Thus, while the Symplectic method preserves energy conservation, the method would be ill-suited to situations where the phase of the system needs to be modelled properly over long timescales.

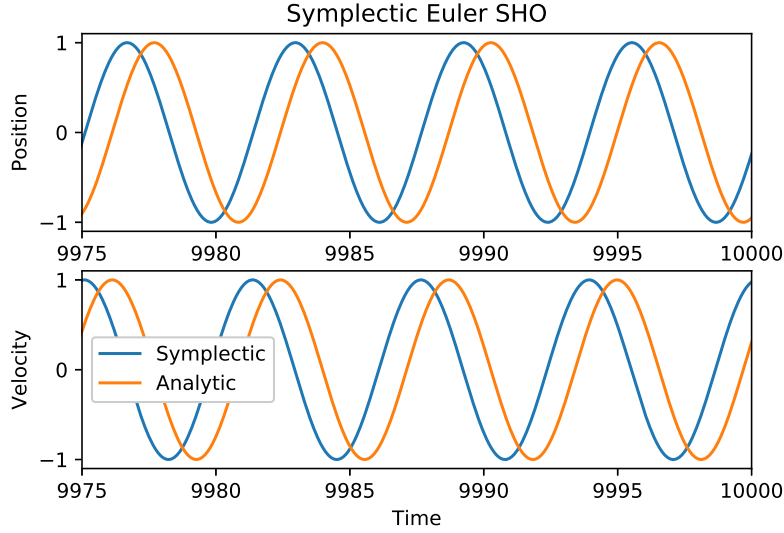


Figure 12: phase error of Symplectic Euler SHO

3 Conclusion

Overall, we can see that each of these first-order methods can model and preserve different aspects of the simple harmonic oscillator system. The suitability of each method for a given application would depend on the tolerance for different types of error (energy vs phase, etc).