

# A Comparative Study of the Performance of Different Numerical Approximations for N-body Problems

Mohammad-Sufian

06 December 2023

## 1 Abstract

The N-body problem is a system of N second-order differential equations that cannot be expressed as a closed-form solution to date. This report weighs the options of using various numerical approximation methods to solve these problems, which are valuable in modelling the early structuring of our universe and the formation of star clusters in astrodynamics. A comparison of methods is made on the basis of accuracy, (computational) efficiency, and stability, which are usually the three quantities of interest in numerical approximation. Preliminary findings from the problem investigation show that even a simple method like symplectic Euler can provide satisfactory simulation results for some elementary N-body problems with acceptable trade-offs in terms of accuracy and computational efficiency.

## 2 Introduction

When Sir Isaac Newton published his Law of Universal Gravitation in the 17<sup>th</sup> century, he used it to adequately describe Kepler's laws of planetary motion. With this knowledge, the problem of locating the positions of two isolated bodies orbiting around one another at any point in time was solved. Naturally, Newton wondered if he could extend this problem to N arbitrary bodies mutually orbiting one another, which gave rise to what is referred to as the N-body problem. Formally, the N-body problem is defined in equation (1), where  $\mathbf{r}_i$  is the position vector of the  $i^{th}$  body:

$$a_i(t) = \frac{d^2 \mathbf{r}_i(t)}{dt^2} = G \sum_{\substack{k=1 \\ k \neq i}}^N \frac{m_k (\mathbf{r}_k(t) - \mathbf{r}_i(t))}{|\mathbf{r}_k(t) - \mathbf{r}_i(t)|^3} \quad (1)$$

The short answer is no. The best we can do is numerically approximate the position of a body in time, which will always come with accuracy trade-offs. There are thousands of choices for numerical approximation techniques in numerical analysis that can be used to tackle the N-body problem. In this study, four methods are chosen, which will highlight the strengths and weaknesses of different methods in terms of accuracy, efficiency, and stability. The study aims to identify the importance of each of these metrics by using software implementations of chosen methods and analysing the simulation results and error plots. The next section in this report will go into detail about these methods and the study setup specifics. This is then followed by a numerical examples section with the results, discussion and concluding remarks.

## 3 Research Method

The study consists of 4 major components.

### 3.1 The Numerical Methods

The numerical methods chosen for the study can be summarised in the following table:

Method	Order	Global Error
Explicit Midpoint Method	Second	$O(h^2)$
Fourth Order Runge-Kutta	Fourth	$O(h^4)$
Symplectic Euler	First	$O(h)$
Velocity Verlet	Second	$O(h^2)$

One important distinction to make here is that Symplectic Euler and Velocity Verlet are symplectic methods that bound energy errors so that they don't accumulate over time during simulations. It can be shown in the numerical examples section that this symplectic property can play an important part in the accuracy of simulation results.

### 3.2 The Test Cases

While the general N-body problem has no known closed form solution, there are a subset of well-known restricted initial value N-body problems for which the solutions are known. We will use two of them in our numerical examples section, namely the figure-8 orbit and a version of Euler's collinear solution.

### 3.3 Running the Simulations

The simulations are coded in Python with heavy reliance on the NumPy package to provide vectorisation and broadcasting of N-length arrays for optimised element-wise operations.

### 3.4 Comparing Results

In order to create a fair comparison for the test metrics across each numerical method, we define the Hamiltonian (total energy in Hamiltonian systems) in equation (2), where  $p_i$  is the momentum of the  $i^{th}$  body and  $q$  is the position vector:

$$H = T + U = \sum_{i=1}^N \frac{\|\mathbf{p}_i\|^2}{2m_i} - \sum_{1 \leq i < j \leq N} \frac{Gm_i m_j}{\|\mathbf{q}_j - \mathbf{q}_i\|} \quad (2)$$

For a fair comparison, we define the log of the fractional Hamiltonian drift in equation (3), where  $\Delta E$  is the change in Hamiltonian at a particular time and  $E$  is the initial Hamiltonian. Now the error is relative.

$$\log_{10} \left( \frac{\Delta E}{E} \right) \quad (3)$$

In the numerical results, we plot the log of the fractional Hamiltonian drift against simulation time.

## 4 Numerical Examples

Referring to the details in the previous section, the numerical examples are as follows:

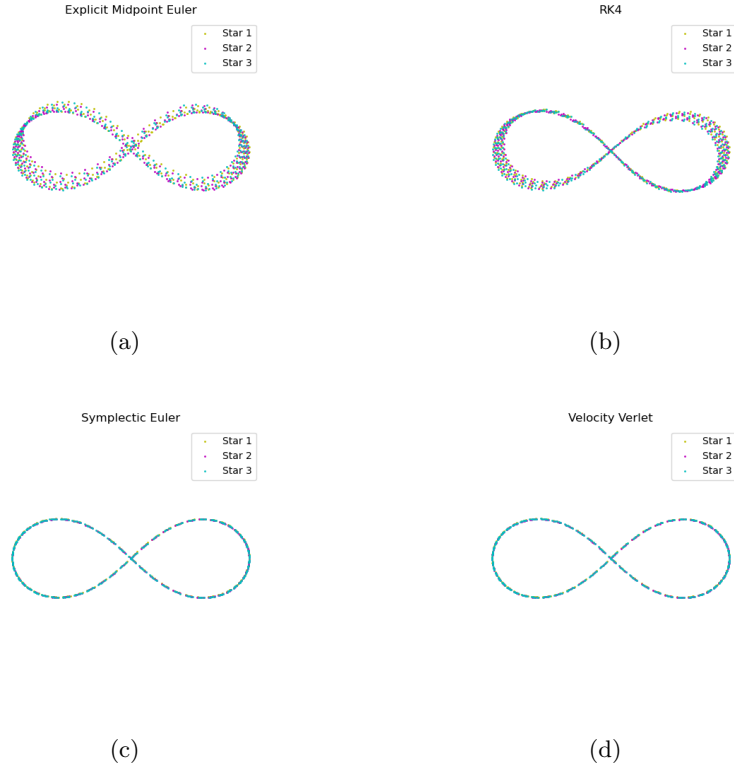


Figure 1: Test Case 1: Figure-8 Orbit Simulations

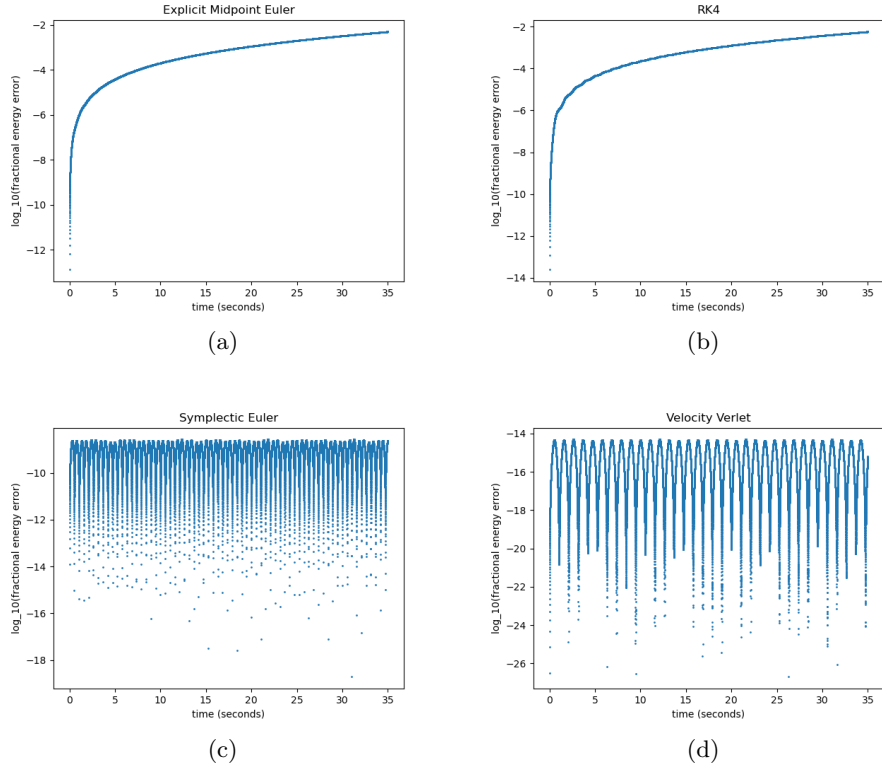


Figure 2: Test Case 1: Figure-8 Orbit Error Plots

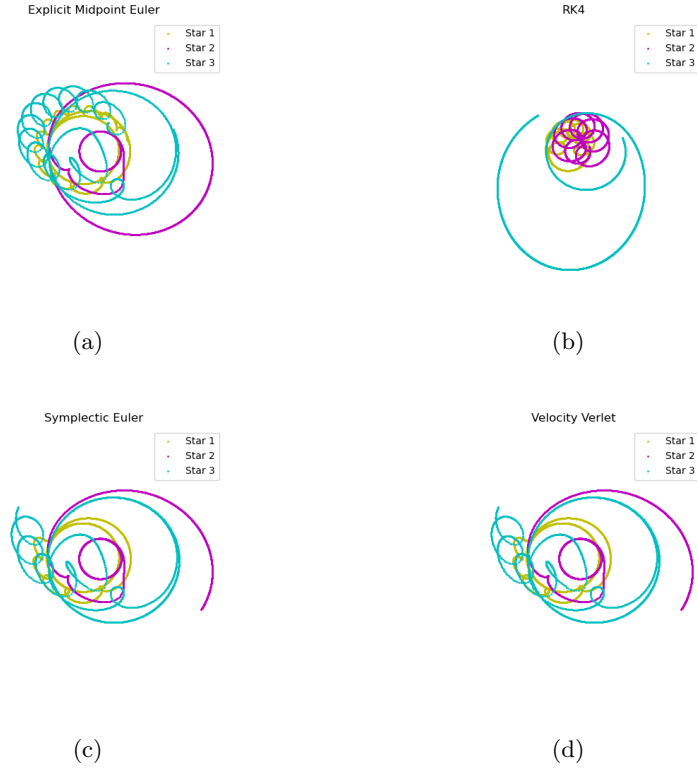


Figure 3: Test Case 2: Chaotic Collinear Configuration Simulations

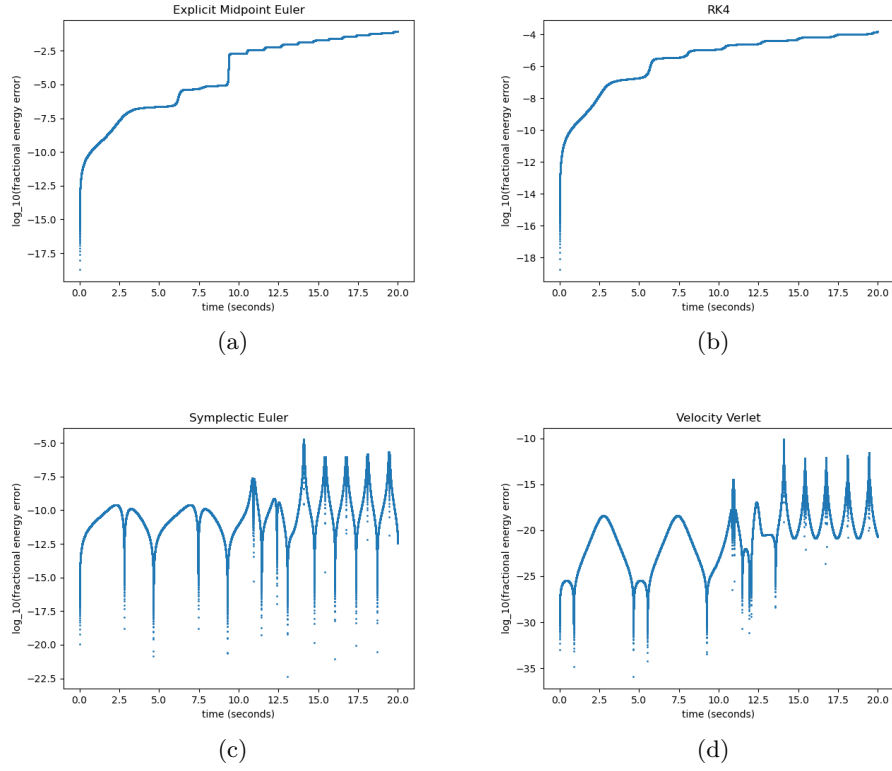


Figure 4: Test Case 2: Chaotic Collinear Configuration Error Plots

## 5 Discussion

### 5.1 Accuracy and Stability

First, consider a more stable case of the figure 8-orbit. When we place three bodies on a figure-8 as far away from each other as mutually possible with a certain initial speed, we expect that the bodies will perpetually orbit one another in a figure-8 shape. As we move across figures 1.a through 1.d, we can start to see a more pronounced figure-8 pattern. This can be explained by the energy plots given in figures 2.a through 2.d. The energy drift for symplectic Euler and Velocity Verlet methods is bounded in an interval, which can preserve the long-term geometry of the figure-8 orbit and hence prevent orbital decay over time, as we can observe from figures 1.c and 1.d. On the other hand, the Explicit Midpoint Method and Fourth Order Runge-Kutta seem to drift increasingly without bound (for our simulation time). If the simulation time were any longer, we might expect even more orbital decay and eventually have bodies drift off. Just from this test case alone, we can conclude that symplecticity is an important factor to consider when choosing to model N-body problems if we want to preserve the long-term geometry of orbits.

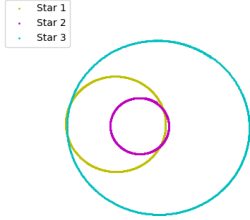


Figure 5: Stable Collinear Configuration

We can observe another test case of a chaotic collinear configuration in figures 3 and 4. This test case is a bit different compared to test case 1. The initial parameters were purposefully tweaked by the same amount to add chaos to the N-body system. Observe the original stable solution in figure 5. We now test the stability of different methods with small perturbations in the initial parameters. There are many criteria to define unstable orbits. In this test case, let's assume that an unstable orbit refers to a high chance of collision between two bodies or if a body is ejected from a system.

In figure 3.a, we can see stars 1 and 3 viciously spiralling towards one another as the simulation progresses. This is usually an indicator of binary stars as they lose energy and approach each other before collision. This probably explains the sharp loss in energy in figure 4.a at around  $t = 9$  seconds. This puts the Explicit Midpoint Method in a bad spot in terms of stability for this test case. While we observe the spiralling effect in the remaining test cases too, the effect is less severe (compare the shapes of 4.a and 4.b, especially the heights where the energy loss ramps up). They do, however, eject bodies from the system eventually, like when we see star 3 leaving the system in 3.b. However, Fourth Order Runge-Kutta is way quicker to eject a body than the symplectic methods to achieve stability quicker.

### 5.2 Efficiency

The computation time of the methods can be summarised in the following table:

Method	Mean Time Per Iteration	Asymptotic Bound
Explicit Midpoint Method	0.2332	$O(N^2)$
Fourth Order Runge-Kutta	0.6930	$O(N^2)$
Symplectic Euler	0.1548	$O(N^2)$
Velocity Verlet	0.2592	$O(N^2)$

The mean time per iteration quite intuitively increases with the order of the method.  $O(N^2)$  is the best achievable bound by brute force since we require information from  $N-1$  bodies to update each of the  $N$  bodies.

## 6 Conclusion

In short, symplecticness is an important property when choosing numerical methods to approximate Hamiltonian systems for preserving long-term geometry. Symplectic Euler provided the fastest computation time and outstanding accuracy at the cost of stability and works well if stable orbit can be ensured. With some loss in efficiency, Velocity Verlet has the promise of being the most accurate and stable numerical method for approximating a large class of  $N$ -body problems in the event of instability.

### 6.1 Limitations

$O(N^2)$  is not an acceptable time complexity when  $N$  gets large. The Barnes-Hut algorithm provides an  $O(N \log N)$  solution compared to the  $O(N^2)$  direct calculation between all pairs of  $N$  bodies.

### 6.2 Future Work

There is great potential for using implicit update methods and variable time steps to improve numerical accuracy.

## References

- [1] K. K. G. MYRDAL, *SOME THEORETICAL AND NUMERICAL ASPECTS OF THE N-BODY PROBLEM*, Faculty of Science, Centre for Mathematical Sciences, Lund University, 2013. <https://lup.lub.lu.se/luur/download?func=downloadFile&recordId=4780668&fileId=4780676>
- [2] O. KOTOVYCH, & J. C. BOWMAN, *An Exactly Conservative Integrator for the n-Body Problem*, Department of Mathematical and Statistical Sciences, University of Alberta, 2002. <https://www.math.ualberta.ca/~bowman/publications/nbody.pdf>
- [3] Z. E. MUSIELAK, & B. QUARLES, *The three-body problem*, Department of Physics, The University of Texas at Arlington, 2015. <https://arxiv.org/pdf/1508.02312.pdf>
- [4] F. P. RAMOS, *Stable Closed Orbits – Figure-8*, School of Mathematics, University of Edinburgh, 2019. [https://www.maths.ed.ac.uk/~ateckent/vacation\\_reports/Report\\_Faustino.pdf](https://www.maths.ed.ac.uk/~ateckent/vacation_reports/Report_Faustino.pdf)
- [5] C. J. VOESENEK, *Implementing a Fourth Order Runge-Kutta Method for Orbit Simulation*, Rochester Institute of Technology, 2008. <http://spiff.rit.edu/richmond/nbody/OrbitRungeKutta4.pdf>
- [6] E. HAIRER, *Symplectic integrators*, Technical University of Munich, 2010. [https://www.unige.ch/~hairer/poly\\_geoint/week2.pdf](https://www.unige.ch/~hairer/poly_geoint/week2.pdf)
- [7] D. M. HERNANDEZ, & E. BERTSCHINGER, *Symplectic integration for the collisional gravitational N-body problem*, Royal Astronomical Society, 2015. <https://doi.org/10.1093/mnras/stv1439>