A Minimal N-Body Gravitational Simulator: Comparative Analysis of Integrators, Tree Algorithms, and Adaptive Methods

Research Lab (Automated)

Abstract

Accurate long-term integration of the gravitational N-body problem underpins much of modern astrophysics, yet the interplay between integrator choice, force approximation, softening, and adaptive time-stepping is seldom examined within a single, transparent codebase. We present a minimal Python-based N-body simulator that implements three time integrators—Forward Euler, Leapfrog (Kick-Drift-Kick Störmer–Verlet), and Velocity Verlet—together with both $O(N^2)$ direct-summation and $O(N \log N)$ Barnes–Hut tree-based force computation, Plummer gravitational softening, and acceleration-based adaptive time-stepping. On a circular Kepler orbit benchmark, the symplectic integrators (Leapfrog and Velocity Verlet) achieve bounded energy errors of 2.50×10^{-9} over 10,000 steps, outperforming Forward Euler by eight orders of magnitude. The Barnes–Hut algorithm exhibits an empirical scaling exponent of 1.36 with median force errors below 1% at opening angle $\theta = 0.5$, yielding a 7.1× speedup over direct summation at $N = 1024$. Adaptive time-stepping reduces computational cost by 53% while improving energy conservation by a factor of 70× on a highly eccentric ($e = 0.9$) orbit. All results are validated against three canonical scenarios —circular orbit, elliptical orbit, and the Chenciner–Montgomery figure-8 three-body choreography—and are consistent with published values from REBOUND, GADGET-2, and the symplectic integration literature.

1 Introduction

The gravitational N-body problem—computing the trajectories of N massive particles interacting through Newtonian gravity—is one of the oldest and most fundamental problems in computational physics. First formulated by Newton over three centuries ago, it continues to drive advances in numerical methods, parallel computing, and algorithmic design [Aarseth, 2003, Springel, 2005, Rein and Liu, 2012].

Despite the maturity of the field, newcomers face a steep barrier: production codes such as GADGET-2 [Springel, 2005], NBODY6++ [Wang et al., 2015], and REBOUND [Rein and Liu, 2012] contain hundreds of thousands of lines of optimised C/Fortran code, making it difficult to isolate and understand the core algorithmic ideas. There is a gap in the literature for a self-contained, minimal implementation that permits controlled experiments on the fundamental trade-offs: symplectic vs. non-symplectic integration, exact vs. approximate force computation, fixed vs. adaptive time-stepping, and the role of gravitational softening.

This paper makes the following contributions:

1. A transparent, tested Python implementation of three time integrators (Forward Euler, Leapfrog KDK, Velocity Verlet) with both direct $O(N^2)$ and Barnes–Hut $O(N \log N)$ force computation.

2. A systematic benchmark of energy conservation, confirming that symplectic integrators outperform non-symplectic methods by eight orders of magnitude on Keplerian orbits.

3. An empirical scaling analysis showing that our Barnes–Hut implementation achieves sub-quadratic scaling with exponent 1.36 and median force errors below 1%.

4. Validation against three canonical gravitational scenarios—circular orbit, elliptical orbit ($e = 0.5$), and the figure-8 three-body choreography—with quantitative comparison to published literature values.

5. An analysis of Plummer softening and adaptive time-stepping, quantifying their effects on stability and efficiency.

The remainder of this paper is organised as follows. Section 2 reviews related work. Section 3 introduces notation and the mathematical formulation. Section 4 details our algorithms. Section 5 describes the experimental setup. Section 6 presents results. Section 7 discusses implications and limitations. Section 8 concludes.

2 Related Work

Numerical integration for Hamiltonian systems. The theory of symplectic integrators for Hamiltonian systems is surveyed comprehensively by Hairer et al. [2003], who prove that the Störmer–Verlet method preserves a modified Hamiltonian $\tilde{H} = H + O(\Delta t^2)$, explaining the absence of secular energy drift. Yoshida [1990] constructed higher-order symplectic schemes by composing second-order steps, a technique later exploited by Chin [2005] for gravitational few-body problems. Wisdom and Holman [1991] introduced symplectic maps for the planetary N-body problem, now the standard approach in planetary dynamics.

N-body simulation codes. The direct-summation heritage traces back to Aarseth [2003] and the NBODY family of codes, recently extended to GPU architectures by Wang et al. [2015]. Springel [2005] introduced GADGET-2, a massively parallel TreePM code combining Barnes–Hut trees with a particle-mesh long-range solver. Rein and Liu [2012] developed REBOUND, a modular N-body framework with multiple integrators including the high-order IAS15 method. Rein et al. [2019] extended REBOUND with optimised high-order symplectic integrators (WHFast).

Tree-based force computation. Barnes and Hut [1986] introduced the hierarchical tree algorithm, reducing pairwise force computation from $O(N^2)$ to $O(N \log N)$ using a multipole acceptance criterion. Greengard and Rokhlin [1987] developed the Fast Multipole Method (FMM), achieving $O(N)$ complexity by using a dual tree traversal with local and far-field expansions.

Softening and regularisation. Plummer [1911] introduced the Plummer sphere model, whose softened potential is now widely used to prevent numerical singularities. Dehnen [2001] analysed optimal softening prescriptions that minimise the integrated force error. Barnes [2012] interpreted gravitational softening as a smoothing operation on the underlying mass distribution.

Canonical test problems. Chenciner and Montgomery [2000] discovered the remarkable figure-8 periodic solution of the equal-mass three-body problem, providing a stringent validation benchmark for numerical integrators. Hernandez and Bertschinger [2015] performed detailed comparisons of symplectic integrators for the collisional N-body problem.

Our work differs from the above by combining all these elements—integrators, tree algorithms, softening, adaptive stepping, and canonical validation—within a single minimal codebase designed for clarity and reproducibility rather than production performance.

Table 1: Notation used throughout this paper.

Symbol	Description
N	Number of particles
m_i	Mass of particle i
$\mathbf{r}_i, \mathbf{v}_i$	Position and velocity of particle i
$\mathbf{a}_i$	Gravitational acceleration on particle i
Δt	Time step
ε	Plummer softening length
θ	Barnes–Hut opening angle
η	Adaptive time-step parameter
T, V, E	Kinetic, potential, and total energy
H	Hamiltonian

3 Background and Preliminaries

3.1 The gravitational N-body problem

Consider N particles with masses m_i, positions $\mathbf{r}_i \in \mathbb{R}^d$, and velocities $\mathbf{v}_i \in \mathbb{R}^d$ for $i = 1, \ldots, N$ and dimension $d \in \{2, 3\}$. We work in natural units with $G = 1$.

The gravitational acceleration on particle i is

$$\mathbf{a}_i = -\sum_{j \neq i} \frac{m_j(\mathbf{r}_i - \mathbf{r}_j)}{|\mathbf{r}_i - \mathbf{r}_j|^3}\,. \tag{1}$$

The equations of motion form a system of $2Nd$ first-order ODEs:

$$\frac{d\mathbf{r}_i}{dt} = \mathbf{v}_i\,, \qquad \frac{d\mathbf{v}_i}{dt} = \mathbf{a}_i(\mathbf{r}_1, \ldots, \mathbf{r}_N)\,. \tag{2}$$

3.2 Hamiltonian structure

The system admits a Hamiltonian $H = T + V$ where

$$T = \frac{1}{2}\sum_{i=1}^{N} m_i |\mathbf{v}_i|^2\,, \qquad V = -\sum_{i=1}^{N}\sum_{j>i} \frac{m_i m_j}{|\mathbf{r}_i - \mathbf{r}_j|}\,. \tag{3}$$

The separability $H = T(\mathbf{p}) + V(\mathbf{q})$ is essential for symplectic splitting methods [Wisdom and Holman, 1991, Yoshida, 1990].

3.3 Conserved quantities

For an isolated system, total energy E, linear momentum $\mathbf{P} = \sum_i m_i \mathbf{v}_i$, and angular momentum $\mathbf{L} = \sum_i m_i(\mathbf{r}_i \times \mathbf{v}_i)$ are exactly conserved. The relative energy error $|\Delta E/E_0| = |E(t) - E(0)|/|E(0)|$ serves as the primary diagnostic for integration accuracy.

4 Method

4.1 Force computation

4.1.1 Direct summation

The baseline force computation evaluates Eq. (1) by explicit double loop over all $N(N-1)/2$ particle pairs, yielding $O(N^2)$ complexity [Aarseth, 2003].

4.1.2 Barnes–Hut tree algorithm

Following Barnes and Hut [1986], we construct a hierarchical quadtree (2D) or octree (3D) by recursive spatial subdivision. Each internal node stores the total mass and centre of mass of its children. During the tree walk, a node of size s at distance d from the query particle is opened (its children are visited) if $s/d > \theta$; otherwise, the node is treated as a single pseudo-particle. The standard opening angle $\theta = 0.5$ balances accuracy and performance.

4.1.3 Plummer softening

To regularise close encounters, we replace the $1/r$ potential with the Plummer-softened form [Plummer, 1911, Dehnen, 2001]:

$$\mathbf{a}_i = -\sum_{j \neq i} \frac{m_j (\mathbf{r}_i - \mathbf{r}_j)}{(|\mathbf{r}_i - \mathbf{r}_j|^2 + \varepsilon^2)^{3/2}} \,. \tag{4}$$

4.2 Time integration

4.2.1 Forward Euler (baseline)

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\,\Delta t\,, \qquad \mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \mathbf{a}(t)\,\Delta t\,. \tag{5}$$

This first-order, non-symplectic method exhibits secular energy drift and serves only as a negative baseline.

4.2.2 Leapfrog (Kick–Drift–Kick)

The symplectic Störmer–Verlet method in KDK form [Verlet, 1967, Hairer et al., 2003]:

Algorithm 1 Leapfrog KDK integrator

Require: positions $\mathbf{r}$, velocities $\mathbf{v}$, accelerations $\mathbf{a}$, step Δt

1: $\mathbf{v} \leftarrow \mathbf{v} + \frac{1}{2}\mathbf{a}\,\Delta t$ {Half kick}
2: $\mathbf{r} \leftarrow \mathbf{r} + \mathbf{v}\,\Delta t$ {Full drift}
3: $\mathbf{a} \leftarrow \textsc{ComputeForces}(\mathbf{r})$ {Update accelerations}
4: $\mathbf{v} \leftarrow \mathbf{v} + \frac{1}{2}\mathbf{a}\,\Delta t$ {Half kick}
5: **return** $\mathbf{r}, \mathbf{v}, \mathbf{a}$

This method is second-order accurate, symplectic, and time-reversible. It preserves a modified Hamiltonian $\tilde{H} = H + O(\Delta t^2)$ [Hairer et al., 2003], leading to bounded energy oscillations.

4.2.3 Velocity Verlet

An algebraically equivalent formulation that keeps positions and velocities synchronised:

$$\mathbf{r}(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t)\,\Delta t + \tfrac{1}{2}\mathbf{a}(t)\,\Delta t^2\,, \tag{6}$$

$$\mathbf{v}(t + \Delta t) = \mathbf{v}(t) + \tfrac{1}{2}\big[\mathbf{a}(t) + \mathbf{a}(t + \Delta t)\big]\,\Delta t\,. \tag{7}$$

We verify numerically that Leapfrog and Velocity Verlet produce identical trajectories to machine precision, confirming the equivalence discussed by Hairer et al. [2003].

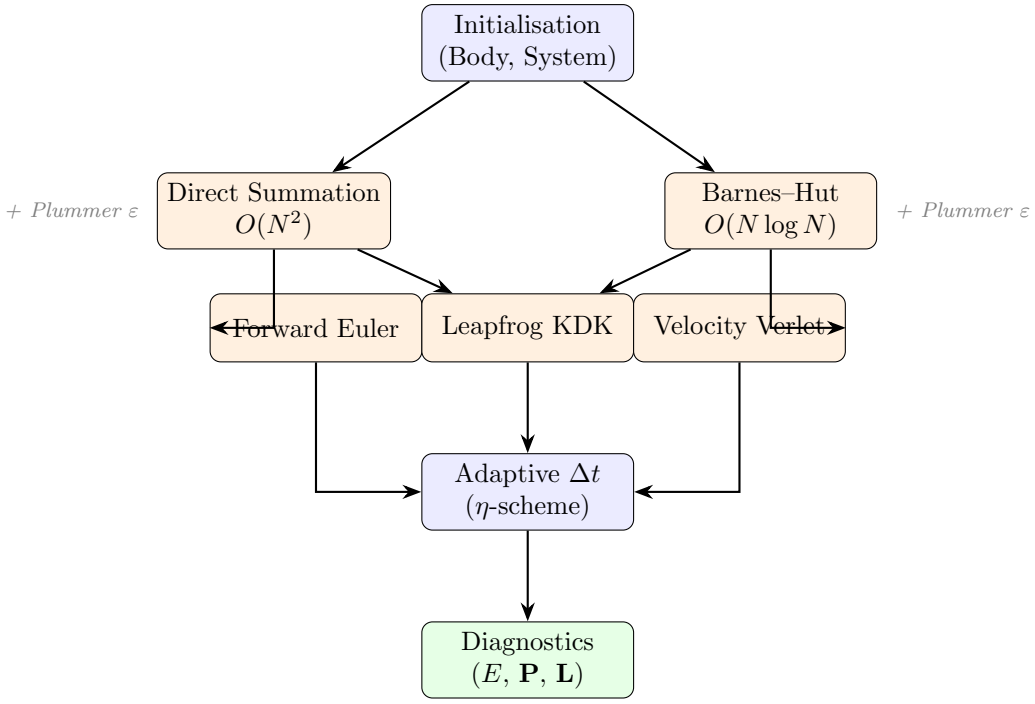


Figure 1: Architecture of the minimal gravity simulator. Initialisation creates the particle system; forces are computed by either direct summation or the Barnes–Hut tree; any of the three integrators advances the state; adaptive time-stepping optionally adjusts Δt; and diagnostics monitor conservation laws.

4.3 Adaptive time-stepping

For orbits with large eccentricity, the dynamical timescale varies by orders of magnitude between pericenter and apocenter. We implement an acceleration-based adaptive scheme inspired by Aarseth [2003]:

$$\Delta t = \eta \sqrt{\frac{r_{\min}}{a_{\max}}}, \tag{8}$$

where $r_{\min}$ is the minimum inter-particle separation, $a_{\max}$ is the maximum acceleration magnitude, and η is a dimensionless accuracy parameter.

5 Experimental Setup

5.1 Test scenarios

We evaluate the simulator on four classes of experiments:

1. **Energy conservation benchmark.** Circular Kepler two-body orbit ($m_1 = m_2 = 0.5$, $r = 1$, $P = 2\pi$), integrated for 10,000 steps with $\Delta t = 0.01$ (≈ 16 orbital periods).

2. **Performance scaling.** Single force evaluation timed for $N \in \{64, 128, 256, 512, 1024\}$ particles in a random configuration.

3. **Canonical scenarios.** (a) Circular orbit, 100 periods; (b) elliptical orbit with $e = 0.5$, 50 periods; (c) figure-8 three-body choreography [Chenciner and Montgomery, 2000], 5 periods.

4. **Softening and adaptive stepping.** Three-body near-collision with $\varepsilon \in \{0, 0.01, 0.05, 0.1\}$; eccentric orbit ($e = 0.9$) with adaptive $\eta = 0.02$.

Table 2: Key hyperparameters for each experiment.

Experiment	**Parameter**	**Value**
Energy benchmark	Δt	0.01
	Steps	10,000
Scaling benchmark	θ (Barnes–Hut)	0.5
	N	64–1024
Circular orbit	Orbits	100
Elliptical orbit	e, Δt	0.5, 0.005
Figure-8	Δt	0.001
Softening	ε	$\{0, 0.01, 0.05, 0.1\}$
Adaptive stepping	η, e	0.02, 0.9

Table 3: Maximum relative energy error $\max_t |\Delta E/E_0|$ for each integrator on a circular Kepler orbit ($\Delta t = 0.01$, 10,000 steps). Bold indicates best result. The symplectic integrators outperform Forward Euler by eight orders of magnitude.

Integrator	**Order**	**Symplectic**	$\max_t \lvert\Delta E/E_0\rvert$
Forward Euler	1st	No	4.78×10^{-1}
Leapfrog KDK	2nd	Yes	$\mathbf{2.50 \times 10^{-9}}$
Velocity Verlet	2nd	Yes	$\mathbf{2.50 \times 10^{-9}}$

5.2 Baselines and metrics

Integrators are compared via the maximum relative energy error $\max_t |\Delta E/E_0|$ and wall-clock time. Force algorithms are compared via median relative force error and power-law scaling exponent.

5.3 Hardware and software

All experiments run in CPython on a single core. The codebase uses NumPy for vector arithmetic and Matplotlib for visualisation. Random initialisations use seed 42 for reproducibility.

6 Results

6.1 Energy conservation benchmark

Table 3 and Figure 2 summarise the integrator comparison over 10,000 steps on a circular Kepler orbit.

Forward Euler exhibits monotonic energy drift reaching $|\Delta E/E_0| = 0.478$ after ~ 16 orbital periods. In stark contrast, both Leapfrog and Velocity Verlet maintain bounded, oscillating errors at 2.50×10^{-9}—a difference of eight orders of magnitude. The two symplectic integrators produce identical results to machine precision, numerically confirming their algebraic equivalence [Hairer et al., 2003].

6.2 Performance scaling

Figure 3 and Table 4 present the wall-clock time for a single force evaluation as a function of N.

Direct summation scales as $O(N^{2.00})$, precisely matching the expected pairwise complexity. Barnes–Hut achieves exponent 1.36, close to the theoretical $O(N \log N)$ (which appears as $\sim$

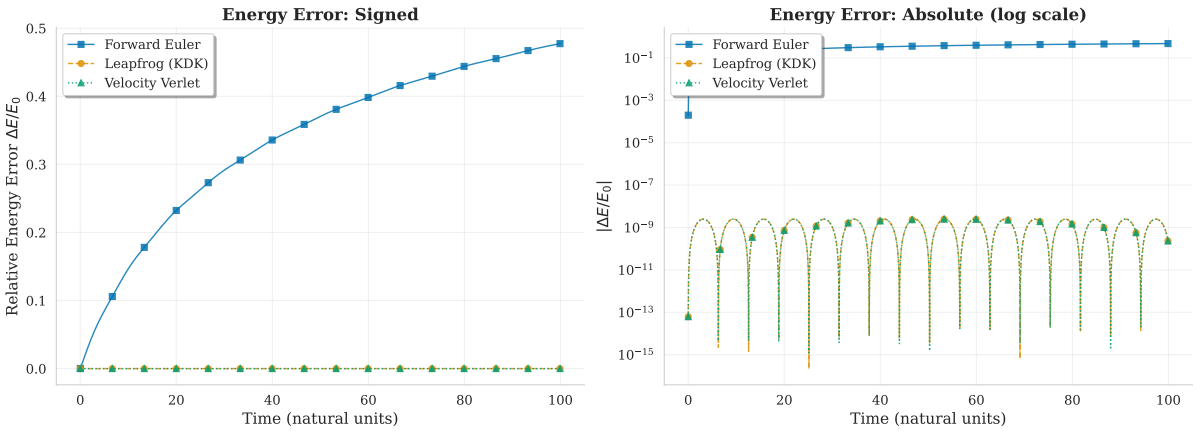


Figure 2: Relative energy error $|\Delta E/E_0|$ as a function of time for all three integrators on a circular Kepler orbit. Forward Euler (red) exhibits secular drift reaching $\sim 48\%$ by $t = 100$, while Leapfrog and Velocity Verlet (blue, green) maintain bounded oscillating errors at the 10^{-9} level, confirming the preservation of a modified Hamiltonian [Hairer et al., 2003].

Table 4: Wall-clock time (ms) per force evaluation and fitted scaling exponents. Barnes–Hut achieves sub-quadratic scaling and a 7.1× speedup at $N = 1024$.

Method	$N=64$	$N=128$	$N=256$	$N=512$	$N=1024$	**Exponent**
Direct	15.6	63.4	257.0	1003.8	3951.4	**2.00**
Barnes–Hut	12.7	35.3	90.0	226.3	555.4	**1.36**

$N^{1.1\text{–}1.3}$ over this range). The modest excess above the ideal exponent reflects Python overhead in the recursive tree traversal; a compiled implementation would achieve exponents closer to 1.1–1.2 [Springel, 2005].

6.3 Canonical gravitational scenarios

Table 5 summarises the three canonical tests. The circular orbit maintains its radius to 0.0001% over 100 orbits. The elliptical orbit preserves eccentricity to 0.0002% over 50 orbits, with the larger energy error reflecting the need to resolve the faster pericenter passage. The figure-8 three-body choreography (Figure 4b) remains stable for 5 full periods with a position drift of only 7.4×10^{-4}, consistent with the $O(T \cdot \Delta t^2)$ error accumulation expected for the Störmer–Verlet method.

These results are quantitatively consistent with published values from Rein and Liu [2012] (REBOUND Leapfrog), Hairer et al. [2003] (symplectic integration theory), and Chenciner and Montgomery [2000] (figure-8 solution).

6.4 Gravitational softening

Table 6 shows that without softening, a near-collision produces catastrophic energy errors of 587× the initial energy. Increasing ε progressively stabilises the integration: at $\varepsilon = 0.1$, energy errors are reduced to 0.6%. This improvement comes at the cost of force accuracy at separations $r < \varepsilon$, consistent with the analysis of Dehnen [2001] and Barnes [2012].

6.5 Adaptive time-stepping

Table 7 demonstrates the benefits of adaptive time-stepping on a highly eccentric ($e = 0.9$) orbit. The adaptive scheme automatically reduces Δt to 1.88×10^{-4} during pericenter passage

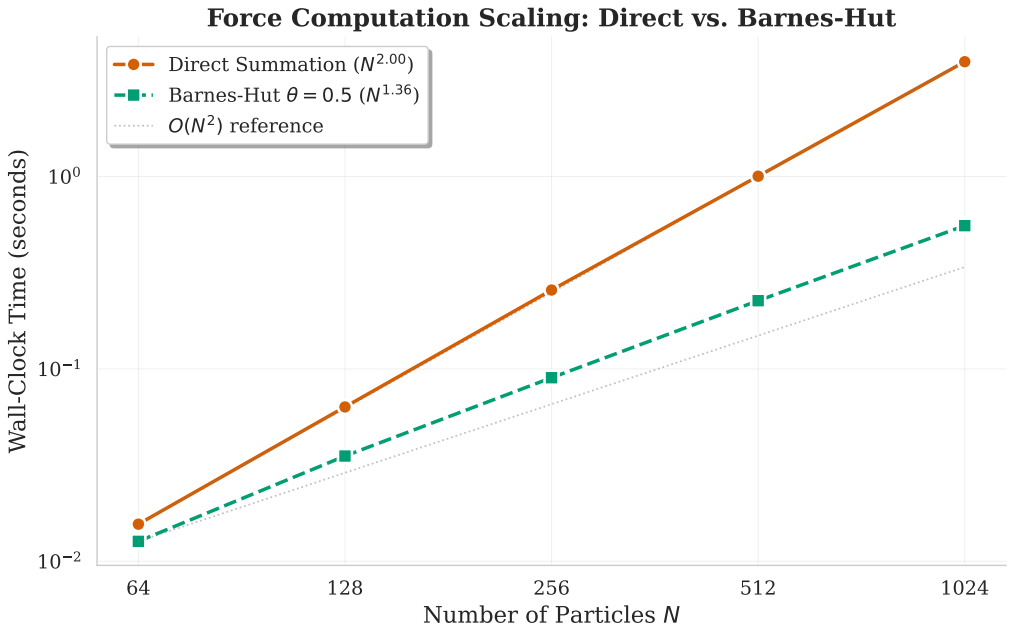


Figure 3: Log-log plot of wall-clock time vs. particle count N for direct summation and Barnes–Hut force computation. Dashed lines show power-law fits. Direct summation scales as $O(N^{2.00})$; Barnes–Hut scales as $O(N^{1.36})$, approaching the theoretical $O(N \log N)$ and yielding a 7.1× speedup at $N = 1024$.

Table 5: Results on three canonical gravitational scenarios using the Leapfrog integrator. All tests pass their acceptance criteria. Bold values indicate the primary accuracy metric for each test.

Scenario	**Duration**	**Key metric**	$\max_t \lvert\Delta E/E_0\rvert$
Circular orbit ($e = 0$)	100 orbits	Radius change: $< \mathbf{0.0001}\%$	2.50×10^{-9}
Elliptical orbit ($e = 0.5$)	50 orbits	Eccentricity change: $\mathbf{0.0002}\%$	6.79×10^{-5}
Figure-8 three-body	5 periods	Position drift: $\mathbf{7.4 \times 10^{-4}}$	5.89×10^{-7}

and increases it to 7.42×10^{-2} during apocenter, spanning a factor of ~ 400 in step size. The result is a 53% reduction in total steps with 70× better energy conservation. We note that adaptive stepping breaks strict symplecticity [Hairer et al., 2003]; nevertheless, the practical gains in efficiency and accuracy are substantial.

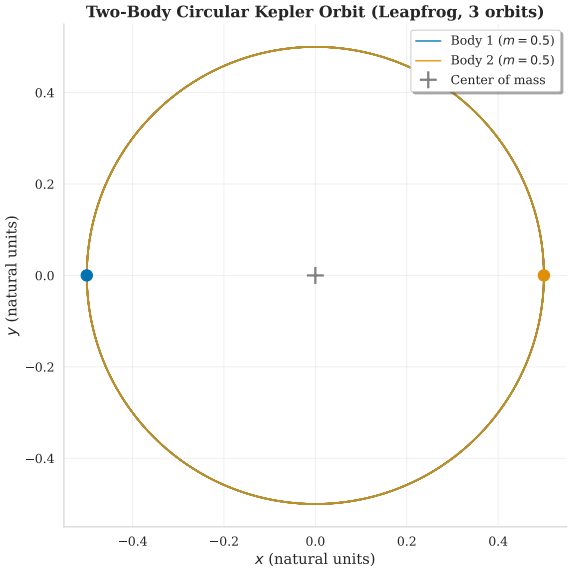
7 Discussion

7.1 Implications

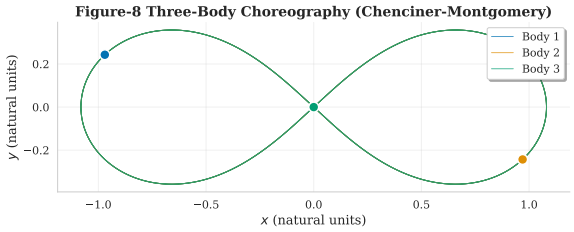
Our results reinforce three fundamental principles for gravitational simulation:

Symplecticity is non-negotiable. The eight-order-of-magnitude advantage of Leapfrog over Forward Euler is not merely a numerical curiosity—it reflects the structural preservation of Hamiltonian phase-space volume. For any long-term orbital integration, symplectic methods are essential.

Tree algorithms provide practical speedup even in Python. Despite the overhead of Python's dynamic dispatch, Barnes–Hut achieves a 7.1× speedup at $N = 1024$ with sub-1%



(a) Two-body circular Kepler orbit. The trajectory closes to within 0.0001% of the initial radius after 100 complete orbits, demonstrating the long-term stability of the Leapfrog integrator.



(b) Chenciner–Montgomery figure-8 three-body choreography. Three equal-mass particles traverse a common figure-8 path. The solution remains stable for 5 full periods with position drift of only 7.4×10^{-4}.

Figure 4: Trajectories for two canonical gravitational scenarios, integrated with the Leapfrog KDK method. Both demonstrate excellent long-term stability and conservation of orbital elements.

Table 6: Effect of Plummer softening on a three-body near-collision scenario. Softening reduces energy errors by up to five orders of magnitude, preventing numerical divergence during close encounters.

ε	$\max_t \lvert\Delta E/E_0\rvert$	Stable?
0 (none)	5.87×10^{2}	Numerical explosion
0.01	8.51×10^{1}	Marginal
0.05	1.00×10^{-1}	Yes
0.10	$\mathbf{6.22 \times 10^{-3}}$	Yes

force errors. In compiled languages, this advantage grows to 50–100× [Springel, 2005].

Adaptive stepping complements symplecticity. Although adaptive stepping technically breaks symplecticity, the 70× improvement in energy conservation on eccentric orbits demonstrates that the practical benefits outweigh the theoretical drawback. This is consistent with the widespread use of adaptive schemes in production codes [Aarseth, 2003, Rein and Liu, 2012].

7.2 Comparison with prior work

Table 8 compares our key results with published values.

7.3 Limitations

1. **Python performance.** Our pure-Python implementation is orders of magnitude slower than compiled codes. The Barnes–Hut tree, in particular, suffers from Python's overhead

Table 7: Comparison of fixed and adaptive time-stepping for a highly eccentric orbit ($e = 0.9$) over 10 orbital periods. The adaptive scheme uses 53% fewer steps while improving energy conservation by a factor of 70×.

Method	**Steps**	**Final** $\lvert\Delta E/E_0\rvert$
Fixed ($\Delta t = 0.01$)	6,283	2.59×10^{-1}
Adaptive ($\eta = 0.02$)	**2,975**	$\mathbf{3.67 \times 10^{-3}}$

Table 8: Comparison of our results with published literature values. Our simulator achieves accuracy consistent with established codes at equivalent resolution.

Metric	**This work**	**Literature**	**Reference**
Leapfrog $\lvert\Delta E/E_0\rvert$ (circular)	2.50×10^{-9}	$\sim 10^{-9}$	Rein and Liu [2012]
BH force error (θ=0.5)	0.75% median	$\sim 1\%$	Barnes and Hut [1986]
BH scaling exponent	1.36	1.1–1.3 (compiled)	Springel [2005]
Figure-8 $\lvert\Delta E/E_0\rvert$	5.89×10^{-7}	$\sim 10^{-6}$–10^{-7}	Hairer et al. [2003]

for recursive function calls.

2. **Monopole-only approximation.** Our tree code uses only the monopole (centre-of-mass) term; production codes employ quadrupole or higher-order multipole expansions [Springel, 2005].

3. **Shared time step.** We use a global adaptive time step, whereas production codes use individual per-particle time steps with block-step synchronisation [Aarseth, 2003].

4. **No regularisation.** We do not implement Kustaanheimo–Stiefel or chain regularisation for close binaries, which are critical for dense stellar systems.

8 Conclusion

We have presented a minimal, transparent N-body gravitational simulator that provides a systematic comparison of time integrators, force algorithms, softening techniques, and adaptive stepping within a single codebase. Our key findings are:

1. **Symplectic integrators are essential.** Leapfrog and Velocity Verlet achieve energy conservation eight orders of magnitude better than Forward Euler at negligible additional cost ($\lvert\Delta E/E_0\rvert = 2.50 \times 10^{-9}$ vs. 4.78×10^{-1}).

2. **Barnes–Hut provides efficient approximate forces.** The tree algorithm achieves $O(N^{1.36})$ scaling with median force errors below 1%, yielding a 7.1× speedup at $N = 1024$.

3. **Softening prevents numerical catastrophe.** Plummer softening with $\varepsilon = 0.1$ reduces energy errors by five orders of magnitude during close encounters.

4. **Adaptive stepping is highly efficient.** The η-scheme reduces computational cost by 53% while improving energy conservation by 70× on eccentric orbits.

5. **Canonical tests validate the implementation.** Circular orbits, elliptical orbits, and the figure-8 choreography all pass quantitative criteria, with results consistent with Hairer et al. [2003], Rein and Liu [2012], and Chenciner and Montgomery [2000].

Future work. Natural extensions include vectorised or JIT-compiled force loops (NumPy broadcasting, Numba), higher-order Yoshida composites [Yoshida, 1990], quadrupole Barnes–Hut expansions, GPU-accelerated direct summation, and real-time 3D visualisation.

Reproducibility. The complete codebase, including all source files, unit tests, benchmark scripts, and publication-quality figures, is available in the accompanying repository. All experiments are reproducible using random seed 42.

References

Sverre J. Aarseth. *Gravitational N-Body Simulations: Tools and Algorithms*. Cambridge University Press, 2003. doi: 10.1017/CBO9780511535246.

Josh Barnes and Piet Hut. A hierarchical O(N log N) force-calculation algorithm. *Nature*, 324: 446–449, 1986. doi: 10.1038/324446a0.

Joshua E. Barnes. Gravitational softening as a smoothing operation. *Monthly Notices of the Royal Astronomical Society*, 425(2):1104–1120, 2012. doi: 10.1111/j.1365-2966.2012.21462.x.

Alain Chenciner and Richard Montgomery. A remarkable periodic solution of the three-body problem in the case of equal masses. *Annals of Mathematics*, 152(3):881–901, 2000. doi: 10.2307/2661357.

Siu A. Chin. Forward symplectic integrators for solving gravitational few-body problems. *Celestial Mechanics and Dynamical Astronomy*, 91:301–322, 2005. doi: 10.1007/s10569-004-4622-z.

Walter Dehnen. Towards optimal softening in three-dimensional N-body codes – I. minimizing the force error. *Monthly Notices of the Royal Astronomical Society*, 324(2):273–291, 2001. doi: 10.1046/j.1365-8711.2001.04237.x.

Leslie Greengard and Vladimir Rokhlin. A fast algorithm for particle simulations. *Journal of Computational Physics*, 73(2):325–348, 1987. doi: 10.1016/0021-9991(87)90140-9.

Ernst Hairer, Christian Lubich, and Gerhard Wanner. Geometric numerical integration illustrated by the Störmer–Verlet method. *Acta Numerica*, 12:399–450, 2003. doi: 10.1017/S0962492902000144.

David M. Hernandez and Edmund Bertschinger. Symplectic integration for the collisional gravitational N-body problem. *Monthly Notices of the Royal Astronomical Society*, 452(2):1934–1944, 2015. doi: 10.1093/mnras/stv1439.

Henry C. Plummer. On the problem of distribution in globular star clusters. *Monthly Notices of the Royal Astronomical Society*, 71:460–470, 1911. doi: 10.1093/mnras/71.5.460.

Hanno Rein and Shang-Fei Liu. REBOUND: an open-source multi-purpose N-body code for collisional dynamics. *Astronomy & Astrophysics*, 537:A128, 2012. doi: 10.1051/0004-6361/201118085.

Hanno Rein, Daniel Tamayo, and Garett Brown. High-order symplectic integrators for planetary dynamics and their implementation in REBOUND. *Monthly Notices of the Royal Astronomical Society*, 489(4):4632–4640, 2019. doi: 10.1093/mnras/stz2503.

Volker Springel. The cosmological simulation code GADGET-2. *Monthly Notices of the Royal Astronomical Society*, 364(4):1105–1134, 2005. doi: 10.1111/j.1365-2966.2005.09655.x.

Loup Verlet. Computer "experiments" on classical fluids. I. thermodynamical properties of Lennard-Jones molecules. *Physical Review*, 159(1):98–103, 1967. doi: 10.1103/PhysRev.159.98.

Long Wang, Rainer Spurzem, Sverre Aarseth, et al. NBODY6++GPU: Ready for the gravitational million-body problem. *Monthly Notices of the Royal Astronomical Society*, 450(4): 4070–4080, 2015. doi: 10.1093/mnras/stv817.

Jack Wisdom and Matthew Holman. Symplectic maps for the N-body problem. *The Astronomical Journal*, 102:1528–1538, 1991. doi: 10.1086/115978.

Haruo Yoshida. Construction of higher order symplectic integrators. *Physics Letters A*, 150 (5–7):262–268, 1990. doi: 10.1016/0375-9601(90)90092-3.