Stellar Dynamics of Cold Collapse, N-Body Simulation

Joshua Benabou

October 9, 2019

Introduction to the N-body problem

N-body problem

Given the initial positions and velocities of N point masses m_i, $i = 1, 2, \ldots, N$ (in an inertial reference frame) which interact gravitationally, determine their trajectories at all future times.

Introduction to the N-body problem

- $N = 2$ (Kepler's problem): 2 bodies move in the same plane in conic sections around their barycenter (Bernoulli,1734).
- $N = 3$: there is no general analytical solution, and the movement is generally non-planar, non-periodic, and chaotic (Poincaré, 1887).
- N large (e.g galaxies, 10^{12} stars): an explicit solution is impossible, we look for a numerical approximation. A **N-body simulation**gives useful information on the *macroscopic* evolution of the system.

Objectives

- **Objective 1**: construct a simulator for $N > 1000$ corps avec un bon compromis entre la précision et le temps de calcul.
- **Objective 2**: modeling of the "cold collapse" of a globular cluster, study of stellar dynamics: evolution towards equilibrium, core collapse, energy exchange via binaries, and evaporation.

Some general properties of an N-body system

From the law of gravitation, $\forall i \in [|1, N|]$:

$$\ddot{\vec{r}}_i = \sum_{i=1, j \neq i}^{N} -Gm_j \frac{\vec{r}_i - \vec{r}_j}{||\vec{r}_i - \vec{r}_j||^3}$$

Potential energy of the system:

$$U = \sum_{1 \leq i < j \leq N} -\frac{Gm_i m_j}{||\vec{r}_i - \vec{r}_j||}$$

Kinetic energy of the system:

$$K = \sum_{i=1}^{N} \frac{1}{2} m_i ||\vec{v}_i||^2$$

For an isolated system, $E = U + K$ is conserved, as well as the momentum of angular momentum.

Simulator construction: code organization

Language: Python.
GUI: Pygame.
Graphs: Matplotlib.
Nb of lines of code: ≈ 1000

We only simulation 2D systems!

Code structure:

-initialization
-Main loop
    1. integration, updating paramétrés des of particle objects
    2. simtime advances one step
    3. data collection
    4. display
-creation of graphs

Simulator construction: choice of integrator

Euler (order 1) gives divergent solutions, even for small time steps.

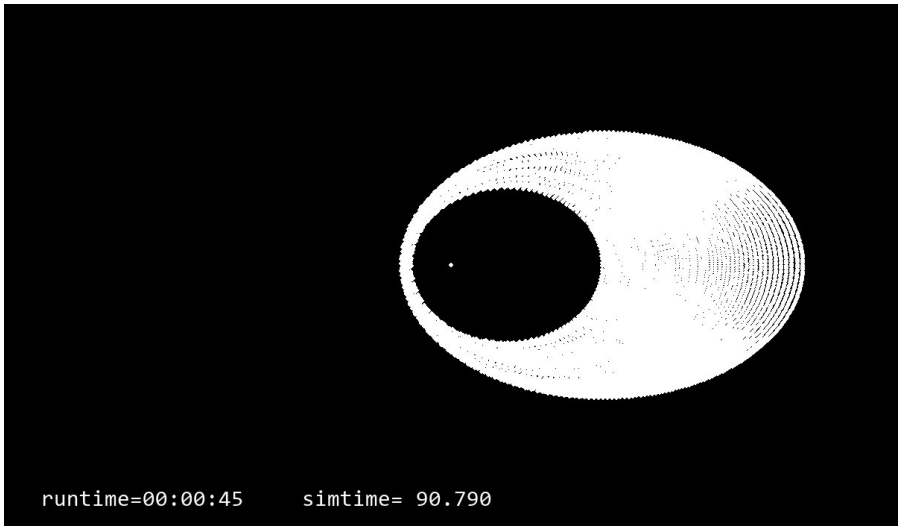
We used Verlet's algorithm:

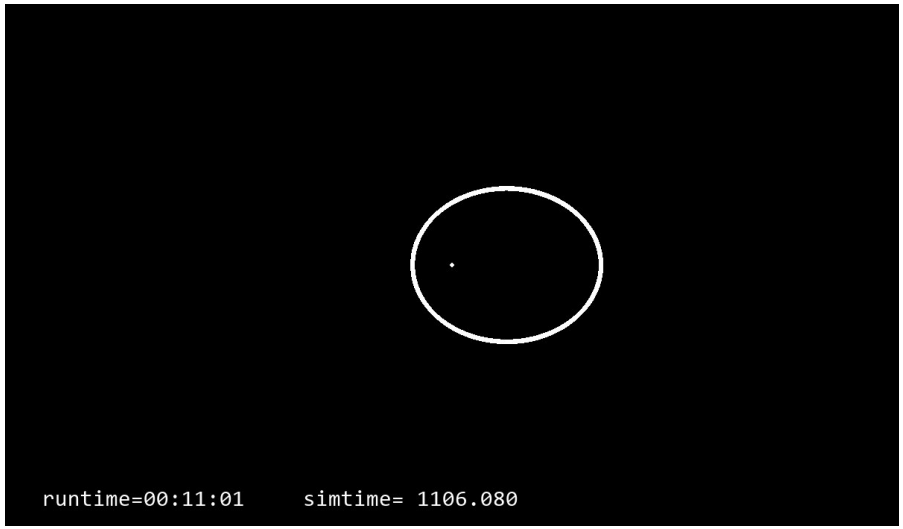
$$\vec{x}(t + \Delta t) = \vec{x}(t) + \vec{v}(t)\,\Delta t + \frac{1}{2}\,\vec{a}(t)\Delta t^2$$

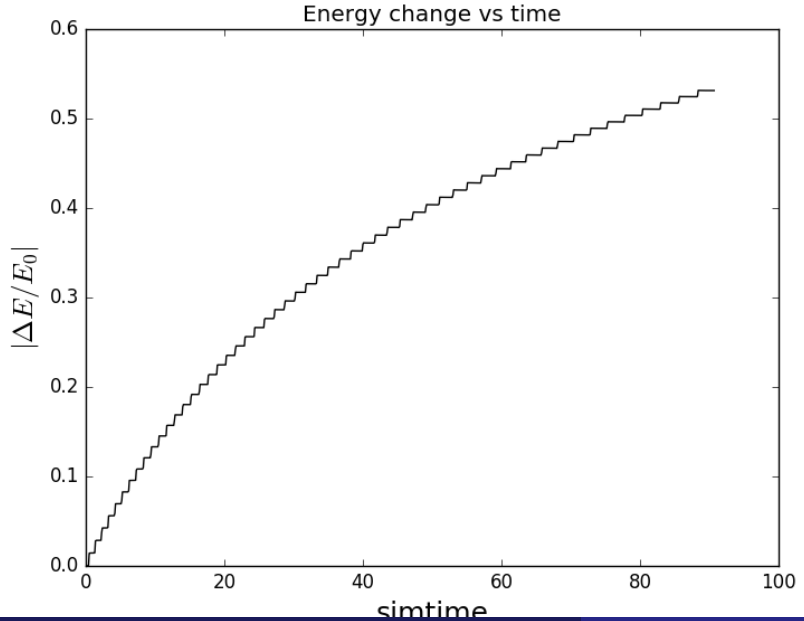
$$\vec{v}(t + \Delta t) = \vec{v}(t) + \frac{\vec{a}(t) + \vec{a}(t + \Delta t)}{2}\Delta t$$

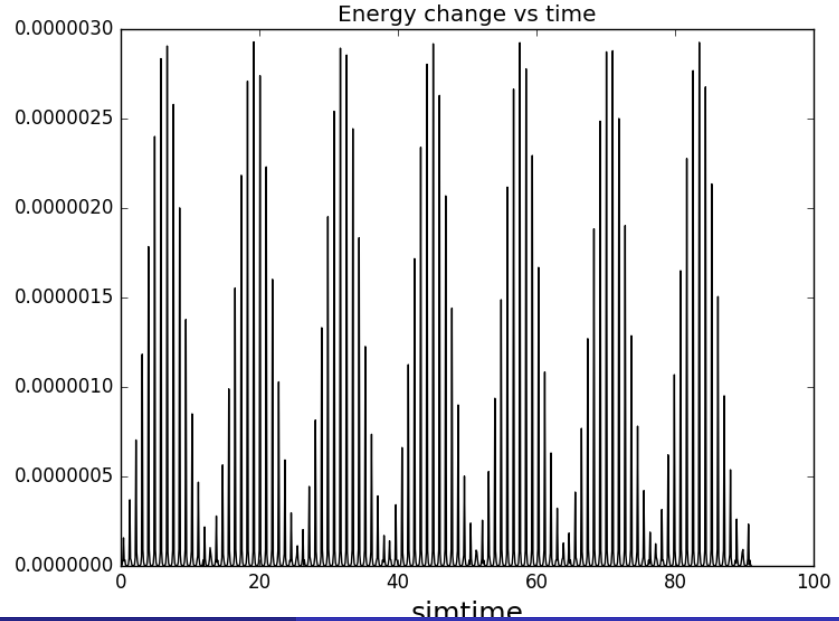
Verlet is second order, time-reversible, and minimizes energy drift.

Simulator construction: choice of integrator









Simulator construction: choice of integrator: "gravitational softening" and units

When the distance between two bodies goes to zero, the force between thme goes to infinity, giving errors.

We use the approximation

$$|F_{i,j}| = \frac{Gm_i m_j}{r^2 + \epsilon}$$

where ϵ is small and chosen to minimize energy drift.

Units: arbitrarily chosen so that the system fits on the screen.

Simulator construction: collisions

In general the simulator doesn't take into account collisions.

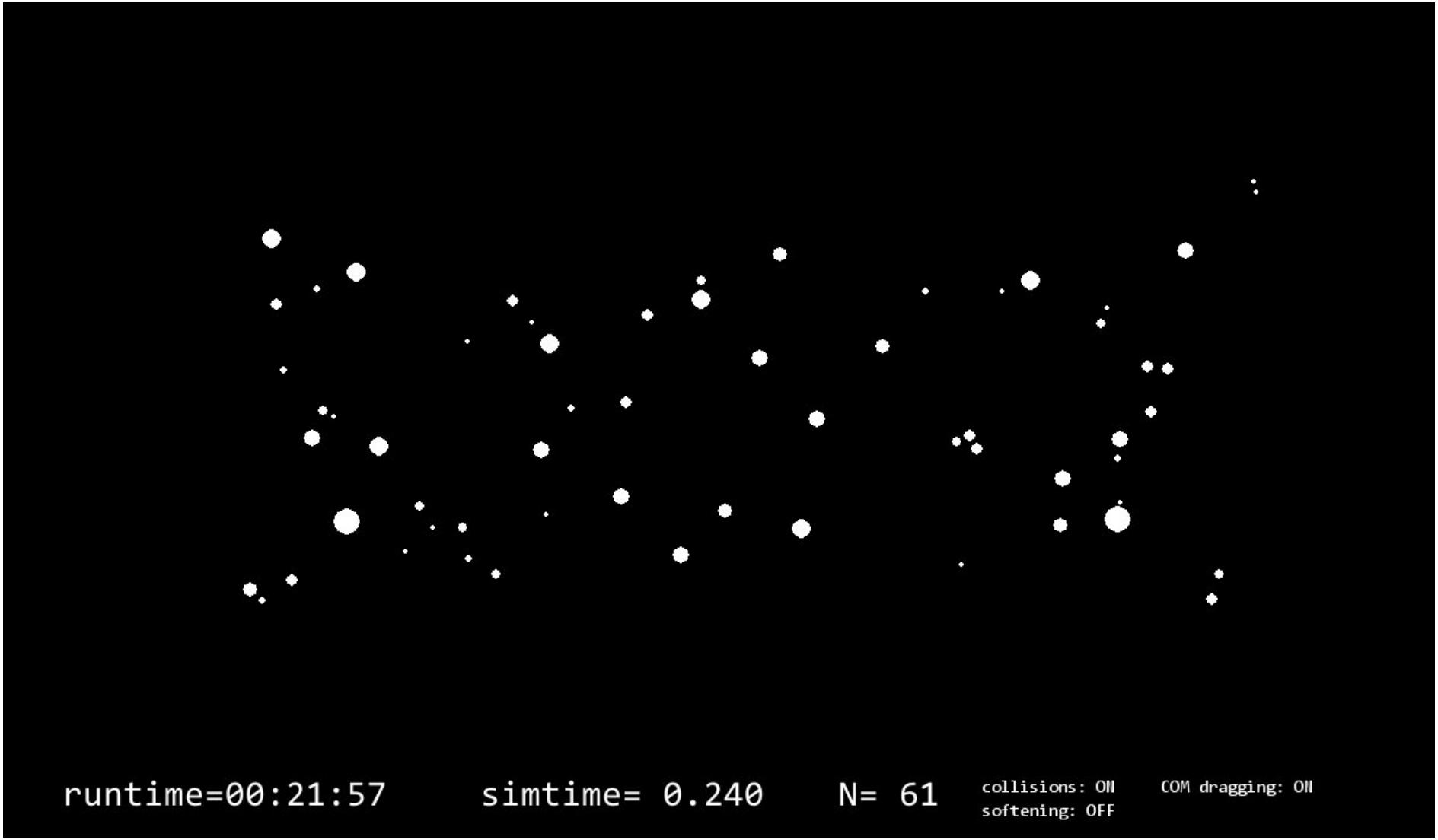
Experiment: simulation of solar system formation

- all the bodies are spherical of the same uniform density
- if two bodies come into contact, they merge
- $N_0 = 200$, particle masses, initial positions and velocities initialized at random.

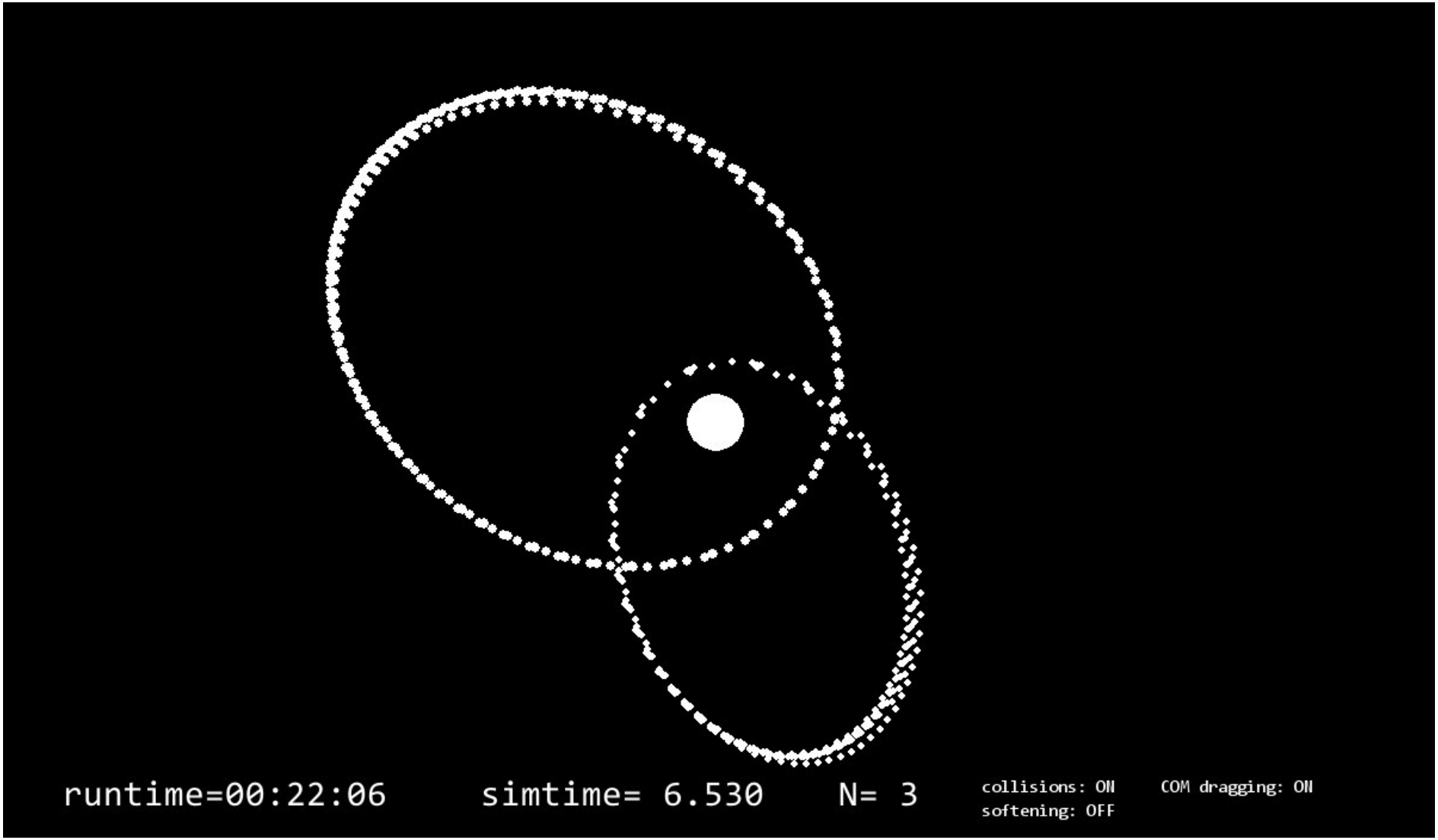
Results: (20 runs)

- In 14 runs, the system is rapidly reduced to a single mass
- In the remaining runs, formation of a "solar system" with between 1 and 5 planets
- In one run, formation of sun-planet-moon system

Simulator construction: collisions



Simulator construction: collisions



Simulator construction: Barnes-Hut optimization

A direct approach requires $\binom{N}{2}$ calculations per time step, complexity is $O(N^2)$.

For $N > 100$ we implemented the **Barnes-Hut** alogirthm, $O(N \ln(N))$, with a minimal loss of precision.

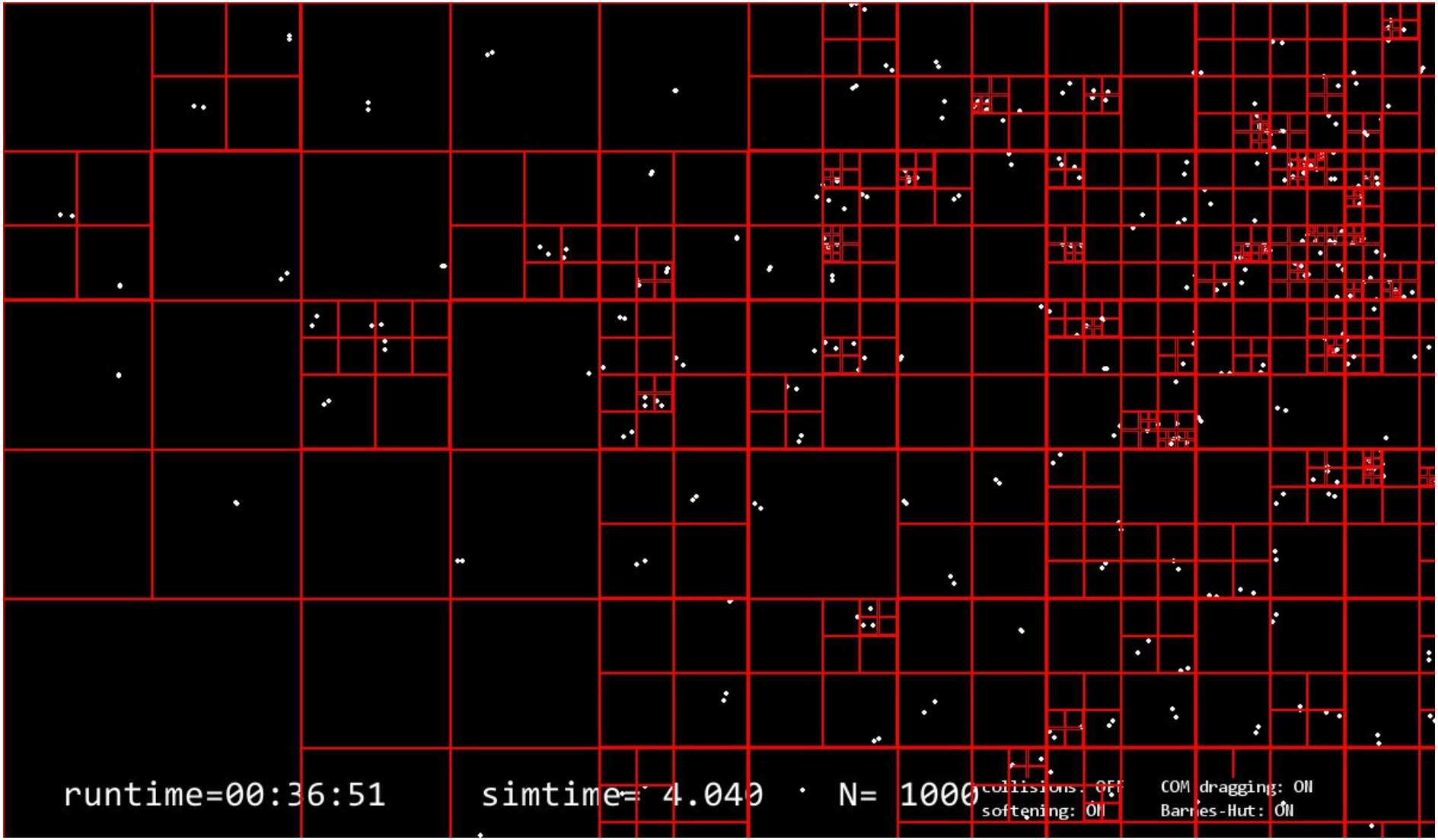
Principal: approximate the force exerted on a particle by a group of bodies close in space by treating them as one body.

The bodies are stored in a quadtree. Each node represents a region of 2D space.

The root represents the whole space, its 4 children the 4 quadrants of this space.

The space is recursively subdivided into quadrants until each subdivision contains 0 or 1 bodies.

Simulator construction: Barnes-Hut optimization



Simulator construction: Barnes-Hut optimization

Each external node represents a single body. Each internal node represents the group of bodies below and stores the barycenter of total mass of each of the bodies of its children.

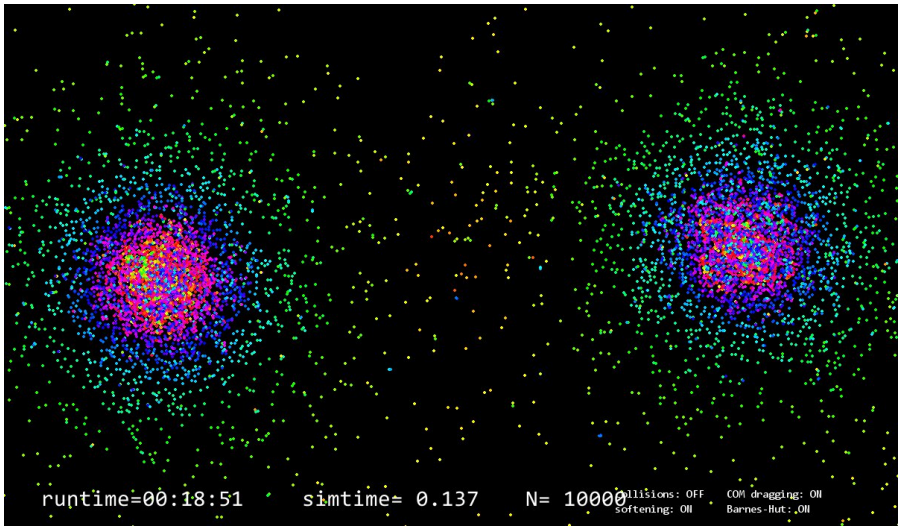
Calculation of force on a body m_i: Iterate through the nodes of tree, starting at the root

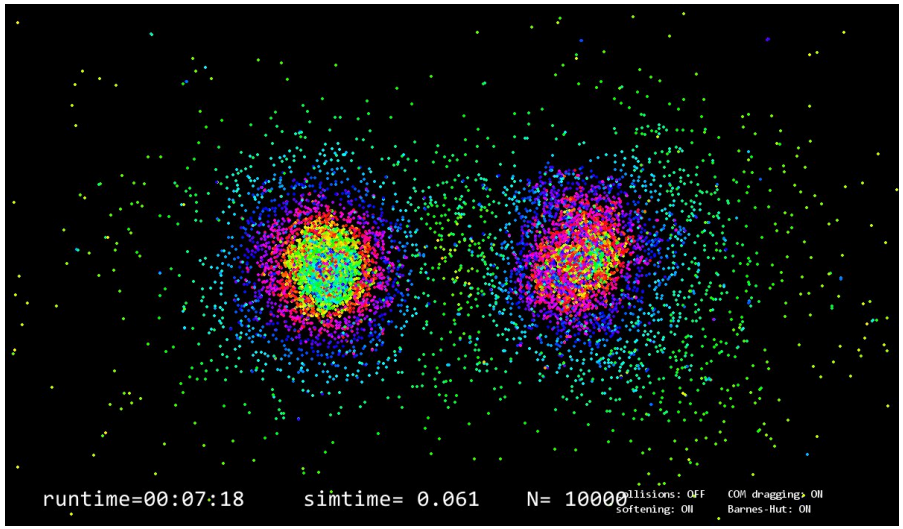
- If the barycenter of an internal node is sufficiently fare from m_i (*), treat the bodies contained in this part of the tree as a single body.
- Otherwise, iterate recursively through each of its subtrees

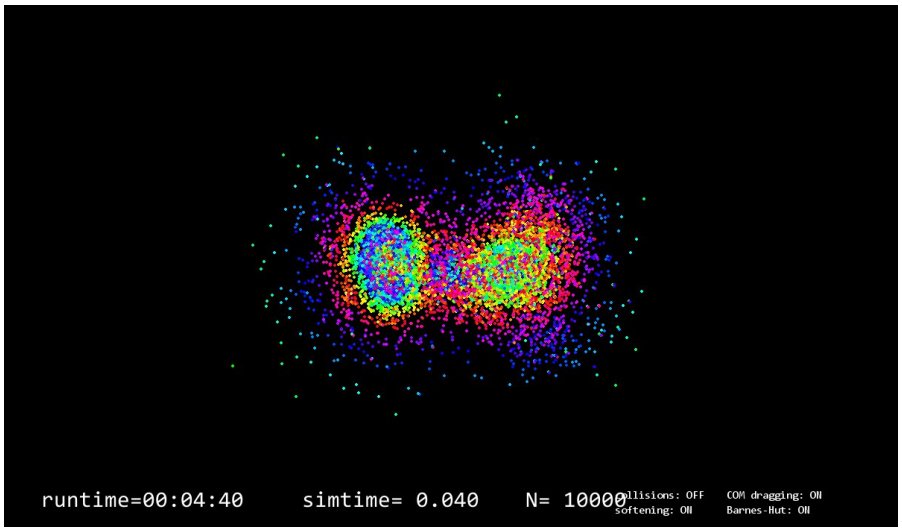
(*)If the angular diameter $s/d < \theta$, où s =size of the region represented by the node, d =distance between m_i and the node's barycenter.

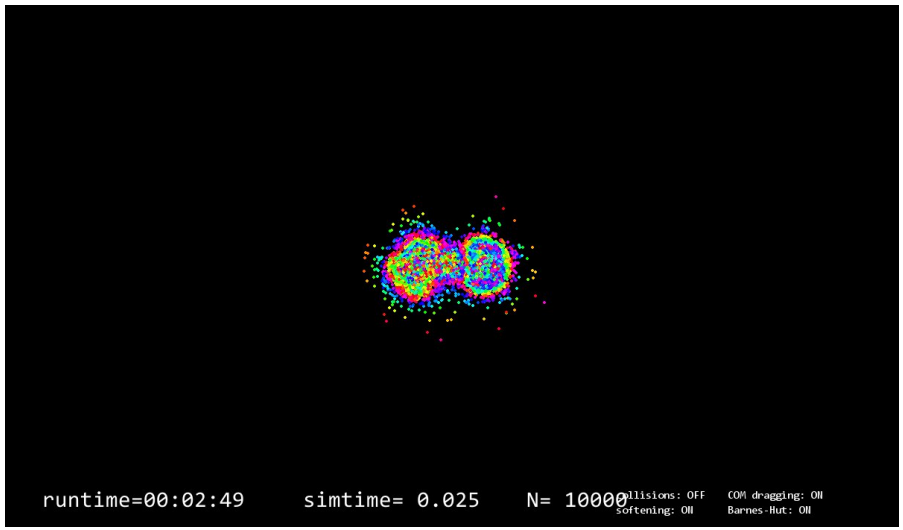
Simulator construction: Barnes-Hut optimization

$N = 10^4$ simulation of collision of two galaxies.

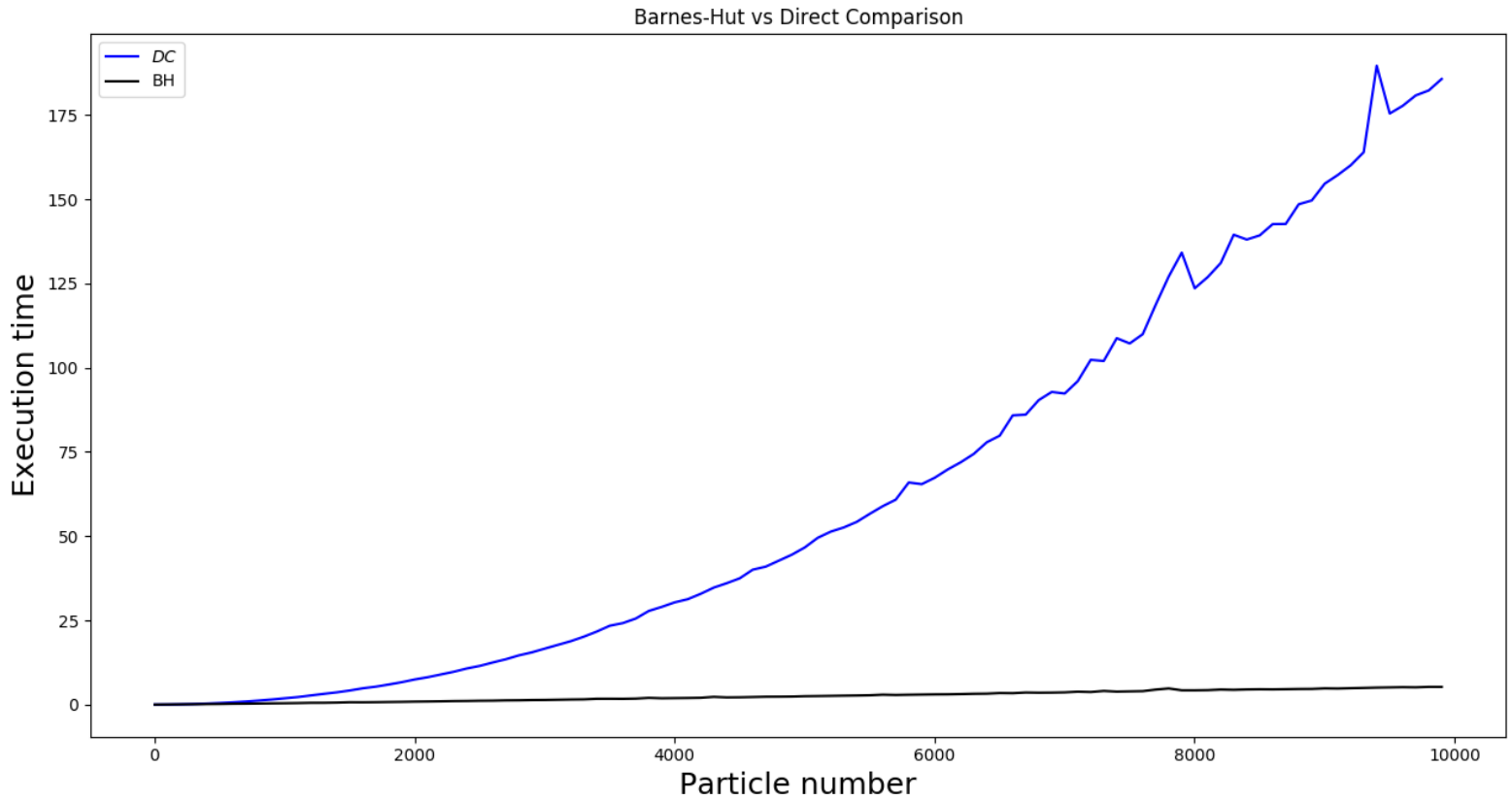








Simulator construction: Barnes-Hut optimization



Stellar dynamics: virial theorem, relaxation

Virial theorem

A *bound* N-body system in equilibrium satifies: $2 < K > + < U >= 0$.

Thus when the system tends towards equilibrium ("relaxation"), $Q = -K/U$ goes to $0,5$.

Mecanism of relaxation in a globular cluster:

- Close encounters between two stars, $\delta v \approx v$ (strong, rare)
- Long range interactions, $b >> r_s$ (weak, frequent)

We can show that the long rane interactions have a greater effect on the movement of a star in the cluster.

Stellar dynamics: relaxation time

relaxation time (t_r) - the average time for the component $v_\perp$ of a star's velocity to be of the same magnitude as its original velocity

Intuitively, the time for a star to lose memory of its original trajectory.

For a system of numerical density n in which all the bodies have equal mass m and average speed V, a probabilistic calculation shows:

$$t_r = \frac{V^3}{8\pi G^2 m^2 n \ln(b_{max}/b_{min})}$$

Example of a globular cluster: $V = 10\ kms^{-1}$, $n = 10^4 pc^{-3}$, $m = M_s$, $t_r = 10^8 an$

Stellar dynamics: consequences of relaxation

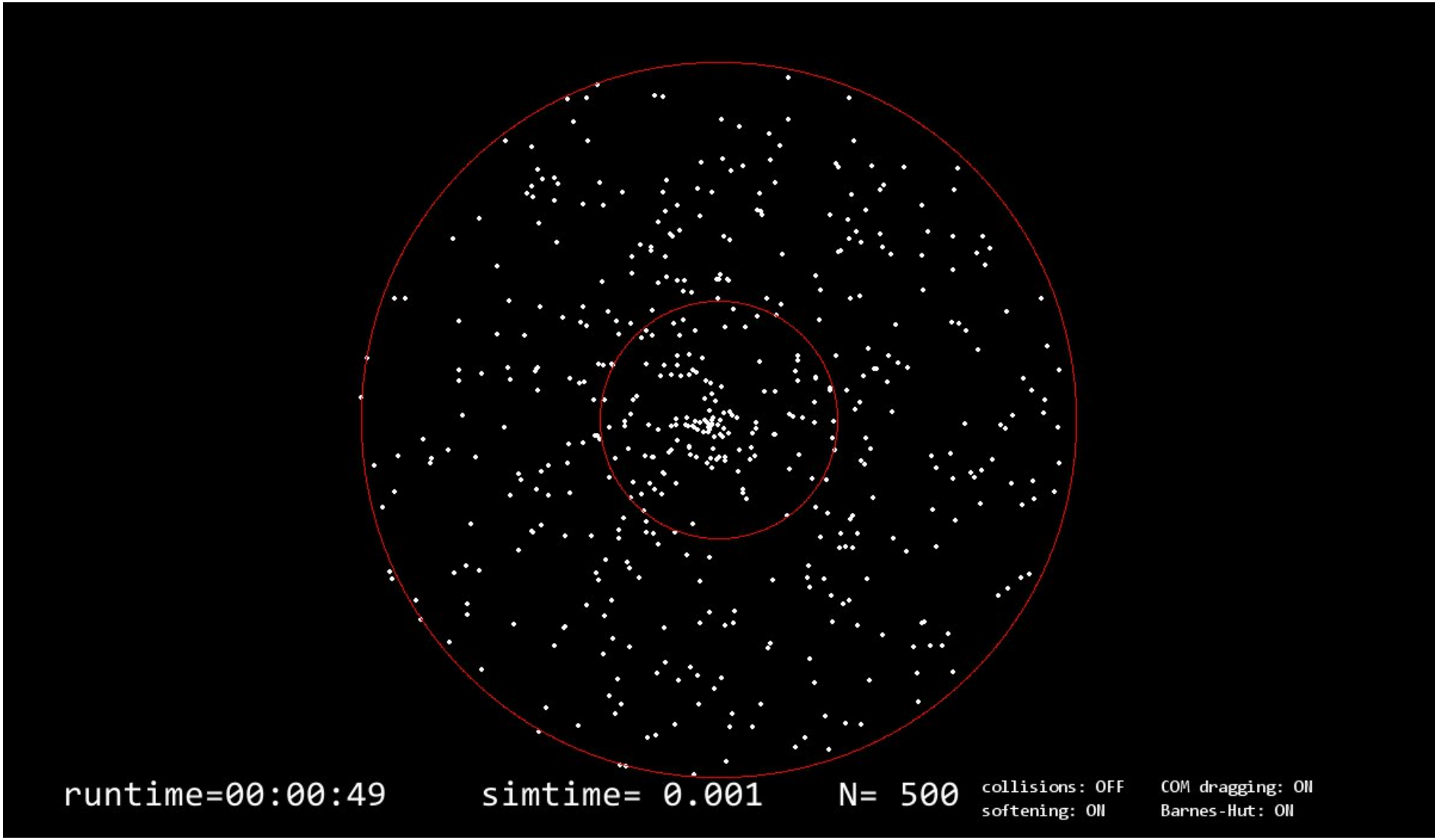
Relaxation creates "conduction" of energy from hot regions (where speeds are high, i.e in the core) towards colder regions (i.e edge)

The stars of the core are cooled down, which makes them fall closer to the center and accelerate; the stars at the edge are heated up, driving them away from the enter of slowing them down.

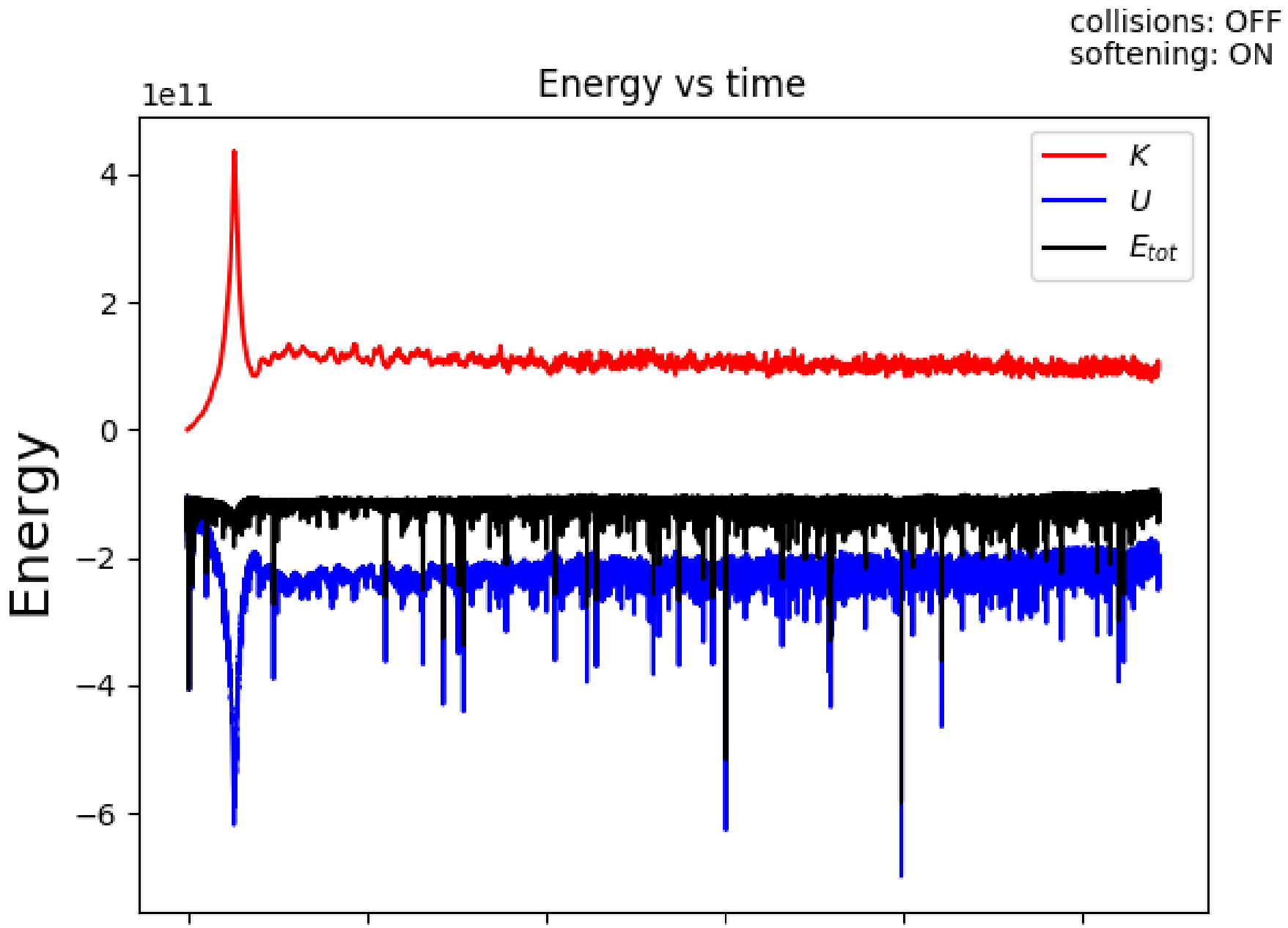
Thus, the temperature gradient increases an the core becomes more compact (time scale$=t_r$)

Evaporation: relaxation allows energy exchange between stars. If the total energy of a star becomes positive, it can leave the cluster ($t_{evap} \approx 100 t_r$)

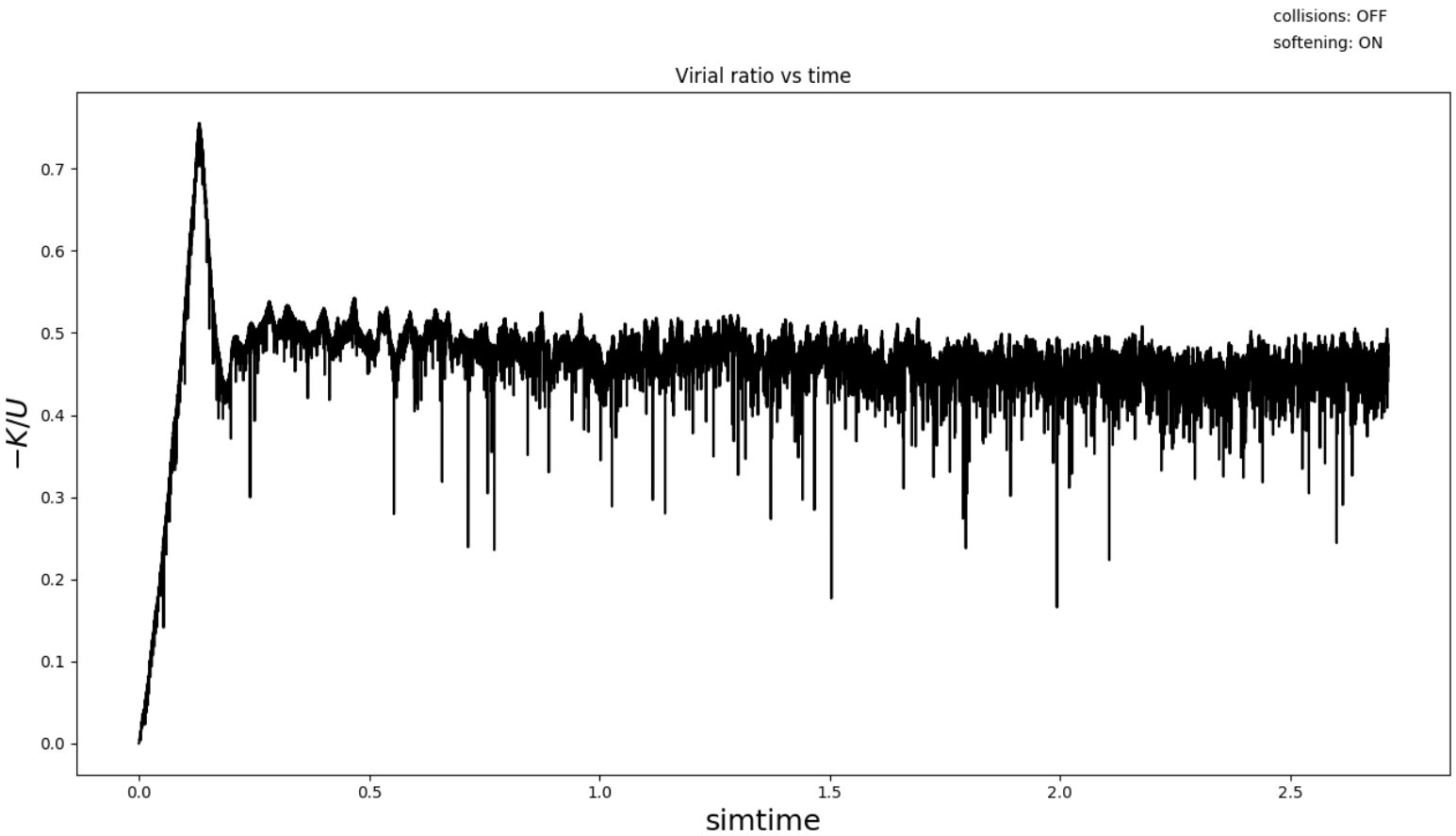
Simulation of cold collapse of a GC



Simulation of cold collapse of a GC



Simulation of cold collapse of a GC



durée= 34h, $< \Delta E/E_0 > = 0.072$, t_r **prédit**=1.29503

Simulation of cold collapse of a GC