N-body techniques for astrophysics

Lecture 7 - Regularization

Mario Spera (mario.spera@oapd.inaf.it) (mario.spera@live.it)
PhD school in Astrophysics,
University of Padova,
November 3-12, 2015

Outline

1. Definition
2. Burdet-Heggie regularization method
   (for the perturbed 2-body problem)
3. Generalization to N-body (hints)

Mathematical background

4. Practical examples

Just practice

General ideas

Key question #1: what is regularization?

Regularizing something means applying several mathematical tricks to transform an equation in a more suitable form for numerical calculations

KEY CONCEPT

Regularizarion helps us to solve mathematical equations in a numerical way

Key question #2: is regularization useful in N-body simulations?

It depends on the system you want to integrate… Regularization works extremely well in case of "strong" gravitational encounters while it isn't really helpful in all the other situations.

MAIN AIM

Understanding WHEN it is worth using regularization and WHY

Regularization

for the perturbed 2-body problem

total mass $M = m_1 + m_2$

relative distance $\boldsymbol{R} = \boldsymbol{R_2} - \boldsymbol{R_1}$

relative velocity $\dot{\boldsymbol{R}} = \boldsymbol{V} = \boldsymbol{V_2} - \boldsymbol{V_1}$

$$\ddot{\boldsymbol{R}} = \boldsymbol{A_{12}} + \boldsymbol{A_{ext}}, \qquad \boldsymbol{A_{12}} = -\frac{GM}{R^3}\boldsymbol{R}$$

Equation of motion (EOM)

2 bodies inside a N-body system

$\boldsymbol{A_{12}}$ isolated 2-body motion

$\boldsymbol{A_{ext}}$ perturbation

acceleration from the other N-2 bodies in the system

$$\boldsymbol{A_1} = \sum_{\substack{j=1 \\ j\neq 1,2}}^{N} \frac{Gm_j}{\left|\boldsymbol{R_j} - \boldsymbol{R_1}\right|^3}\left(\boldsymbol{R_j} - \boldsymbol{R_1}\right) \qquad \boldsymbol{A_{ext}} = \boldsymbol{A_2} - \boldsymbol{A_1}$$

Regularization

for the perturbed 2-body problem

Equation of motion (EOM)

2 bodies inside a N-body system

$$\ddot{\boldsymbol{R}} = \boldsymbol{A_{12}} + \boldsymbol{A_{ext}}, \qquad \boldsymbol{A_{12}} = -\frac{GM}{R^3}\boldsymbol{R}$$

In case of *DOMINANT* 2-body motion $\gamma = \dfrac{|A_{\text{ext}}|}{\frac{M}{R^3}} \ll 1$

$\boldsymbol{A_{12}}$ Responsible for numerical issues

Ultraviolet (UV) divergence

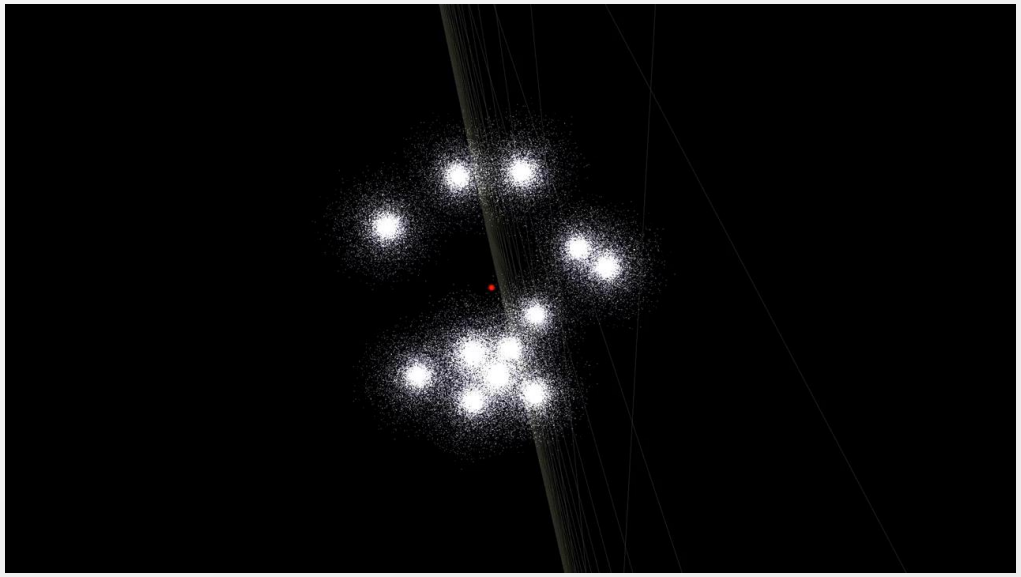
$$R \to 0 \implies A_{12} \implies \infty$$

KEY CONCEPT

The equation of motion is singular for $\boldsymbol{R \to 0}$

Effects of UV divergence

12 globular clusters + 1 Supermassive Black Hole (red spot)



Sooner or later, 2 particles will suffer a

strong gravitational encounter

dominant 2-body motion → UV divergence

Hermite 4th order behavior → TIME STEP REDUCTION

$$\Delta t \propto R^{\alpha}\,, \alpha > 0 \implies \text{if } R \to 0 \implies \Delta t \to 0$$

Numerical integration → SLOW AND INACCURATE

smaller Δt → more steps to reach the same time → larger numerical errors

Let's try to regularize the EOM performing a coordinate transformation in which the expression is not singular anymore in $R = 0$!!!!

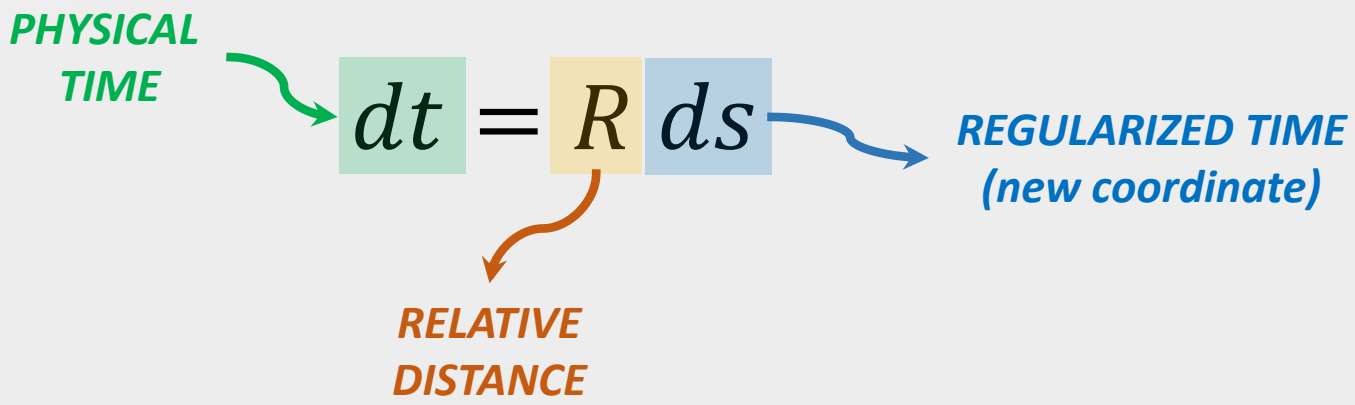
Burdet-Heggie Regularization

for the perturbed 2-body problem

$$\ddot{\boldsymbol{R}} = -\frac{GM}{R^3}\boldsymbol{R} + \boldsymbol{A}_{\mathbf{ext}}$$

EOM
stage 0

Infinitesimal coordinates transformation



$$\dot{\boldsymbol{R}} = \frac{d\boldsymbol{R}}{dt} = \frac{d\boldsymbol{R}}{ds}\frac{ds}{dt} = \boldsymbol{R}'\frac{1}{R} \quad \Longrightarrow \quad \ddot{\boldsymbol{R}} = \frac{d\dot{\boldsymbol{R}}}{dt} = \frac{d}{dt}\left(\boldsymbol{R}'\frac{1}{R}\right) = \frac{1}{R^2}\boldsymbol{R}'' - \frac{R'}{R^3}\boldsymbol{R}'$$

Burdet-Heggie Regularization

for the perturbed 2-body problem

$$\boldsymbol{R}'' = \frac{R'}{R}\boldsymbol{R}' - \frac{GM}{R}\boldsymbol{R} + R^2\boldsymbol{A}_{\mathbf{ext}}$$

EOM stage 1

*Let's introduce an **extremely** important quantity*

Laplace-Runge-Lenz vector

$$\boldsymbol{e} = \frac{\dot{\boldsymbol{R}} \times \left(\boldsymbol{R} \times \dot{\boldsymbol{R}}\right)}{GM} - \frac{\boldsymbol{R}}{R}$$

- ✓ **Constant of motion** (if $\boldsymbol{A}_{\mathbf{ext}} = \mathbf{0}$)
- ✓ Length = **eccentricity**
- ✓ It points toward the pericentre

$$GM\boldsymbol{e} = \dot{\boldsymbol{R}} \times \left(\boldsymbol{R} \times \dot{\boldsymbol{R}}\right) - GM\frac{\boldsymbol{R}}{R} = \frac{|R'|^2}{R^2}\boldsymbol{R} - \frac{R'}{R}\boldsymbol{R}' - \frac{GM}{R}\boldsymbol{R} = \boldsymbol{K}$$

Burdet-Heggie Regularization

for the perturbed 2-body problem

$$\boldsymbol{R}'' = \frac{|\boldsymbol{R}'|^2}{R^2}\boldsymbol{R} - 2\frac{GM}{R}\boldsymbol{R} - \boldsymbol{K} + R^2\boldsymbol{A}_{\mathbf{ext}}$$

EOM stage 2

*Let's introduce another **extremely** important quantity*

Specific orbital energy

$$E_s = \frac{1}{2}\left|\dot{R}\right|^2 - \frac{GM}{R}$$

$$\frac{\textbf{total energy}}{\textbf{reduced mass}}$$

reduced mass = $\mu = \frac{m_1 m_2}{M}$

$$E_s = \frac{1}{2}\left|\dot{R}\right|^2 - \frac{GM}{R} = \frac{1}{2}\frac{|\boldsymbol{R}'|^2}{R^2} - \frac{GM}{R} \Longrightarrow 2E_s\boldsymbol{R} = \frac{|\boldsymbol{R}'|^2}{R^2}\boldsymbol{R} - 2\frac{GM}{R}\boldsymbol{R}$$

Burdet-Heggie Regularization

for the perturbed 2-body problem

$$\boldsymbol{R''} = 2E_s\boldsymbol{R} - \boldsymbol{K} - R^2\boldsymbol{A}_{\mathbf{ext}}$$

Regularized EOM

(not singular anymore in $R = 0$)

Simplification $\boldsymbol{R''} = 2E_s\boldsymbol{R} - \boldsymbol{K}$

CONSTANTS OF MOTION

Regularized EOM for an isolated binary system

Harmonic oscillator subject to a constant force
(Universal Harmonic Oscillator, UHO)

KEY RESULT

The UHO has no singularities → the numerical integrator does not have to reduce the (regularized) time step when $R \to 0$

Regularization, price to pay

Key question #3: since we are using the regularized time **s**, how can we go back to PHYSICAL time **t**?

Form the definition of regularized time we can evaluate, step by step, the corresponding physical time. Nevertheless, the relation between $\boldsymbol{t}$ *and* $\boldsymbol{s}$ *is infinitesimal so we need to solve an integral through an APPROXIMATE method.*

Going from regularized time $\boldsymbol{s_0}$ *to* $\boldsymbol{s_1}$ ***(time step*** $\boldsymbol{\Delta s = s_1 - s_0}$***)***

$$dt = R\,ds \implies \int_{t_0}^{t_1} dt = \int_{s_0}^{s_1} R(s)\,ds \implies \boldsymbol{\Delta t = \int_{s_0}^{s_1} R(s)\,ds}$$

Trapezoidal rule

$$\boldsymbol{\Delta t = \int_{s_0}^{s_1} R(s)\,ds \cong \frac{R(s_0) + R(s_1)}{2}\Delta s + O(\Delta s^3)}$$

We pay a ***time error*** *depending on the method you use to approximate the integral relation between* **t** *and* **s**

Key question #4: is the phase error a real problem for N-body simulations?

For the overwhelming majority of the N-body simulations we are fine with it!!

Mikkola's Algorithmic Regularization -- hints

Sometimes the dominant 2-body motion approximation is not valid → **Generalization to N-body**

KEY IDEA → Generalize the function that generates the coordinates change

$$dt = R\, ds \longrightarrow dt = g(\boldsymbol{q}, \boldsymbol{p}; t)\, ds$$

$$g(\boldsymbol{q}, \boldsymbol{p}; t) = \frac{1}{U(R_{ij})} \qquad U(R_{ij}) = \sum_{i<j}^{N} \frac{m_i m_j}{R_{ij}}$$

N-body gravitational potential

$$dt = \frac{ds}{U(R_{ij})}$$

Time transformation of the Mikkola's algorithmic regularization

Try it yourself

Open terminal → Ctrl + Alt + t

git clone https://gitlab.com/mario.spera/nbodyclass2015.git

This will (should) create a folder named **nbodyclass2015**

Go inside that folder → cd nbodyclass2015

What is inside ?

1) Burdet-Heggie regularization (isolated binary system)
(toy code)

$$\boldsymbol{R''} = 2E_s\boldsymbol{R} - \boldsymbol{K}$$

2) Mikkola's algorithmic regularization (N-body)
(very advanced code :D)

$$dt = \frac{ds}{U(R_{ij})}$$

3) Standard Hermite 4th order integrator with block time steps
(by S. Aarseth… but slightly modified)
http://www.ast.cam.ac.uk/~sverre/web/pages/nbody.htm

All the codes have G = 1

Simple test with B-H regularization

2 stars, equal masses, circular orbit

Key question #5: How we set up initial conditions?

Easy if we remember the following statement: ***a 2-body problem is completely equivalent to considering the motion of a test particle attracted by a FIXED center with mass equal to the TOTAL MASS of the system***

Easiest approach: one particle always at (0, 0, 0) with velocity (0, 0, 0)

Second particle at (R, 0, 0) with velocity (0, V, 0)

If $R = R_0$ then $V_{\text{circular}} = \sqrt{\frac{GM}{R_0}}$

The codes will automatically scale the initial conditions to the **centre of mass reference frame**

Simple test with B-H regularization

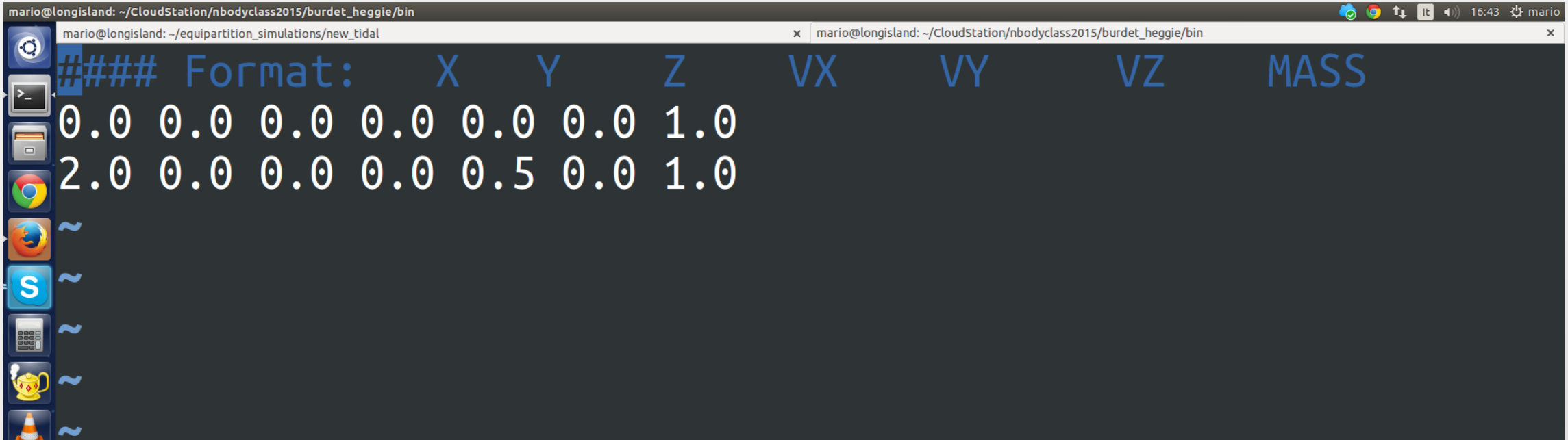
Example $m_1 = m_2 = 1.0$ $R_1 = (0.0, 0.0, 0.0)$ $R_2 = (2.0, 0.0, 0.0)$

$V_1 = (0.0, 0.0, 0.0)$ $V_2 = (0.0, 1.0, 0.0)$

How to let the code read initial conditions → place them in the file **initial_conditions.dat**

cd ./nbodyclass2015/burdet_heggie/bin

nano initial_conditions.dat (modify the file with your initial conditions)

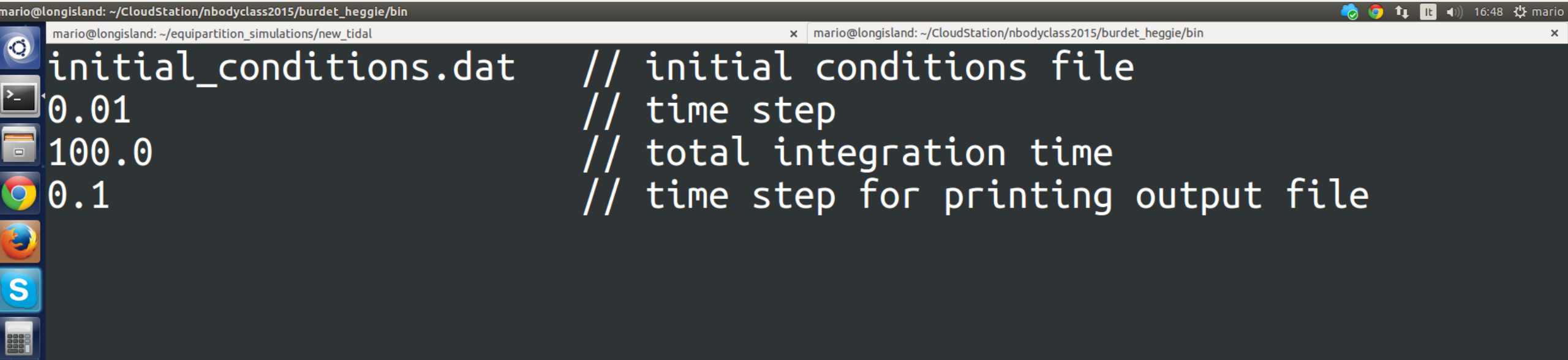


Simple test with B-H regularization

Now, in the same folder, open the file **parameters.txt**

FIXED (regularized) time step here… best value? TRY !!! In general, something around 0.01 times the orbital period (P) is fine ! (The orbital period for our example is 4π….. CHECK!!!)

Every 0.1 time units we have an output… the code will print out
→ positions and velocities (file **output.dat**),
→ relative energy variation (third column) as a function of time (first column) (file **energy.dat**)…
(Second column of energy.dat is the value of the total energy)



Simple test with B-H regularization

compile (just once) and **run**

cd ./nbodyclass2015/burdet_heggie/

make clean (clean all files from previous compilations)
make (compile)
cd bin (folder where you can find the executable)
./burdetheggie.x (execute the code)

Output

```
Eccentricity = 0.000000000000e+00
Time reached by integration = 1.000199969508e+02
Required stopping time      = 1.000000000000e+02
SUCCESS !!
```

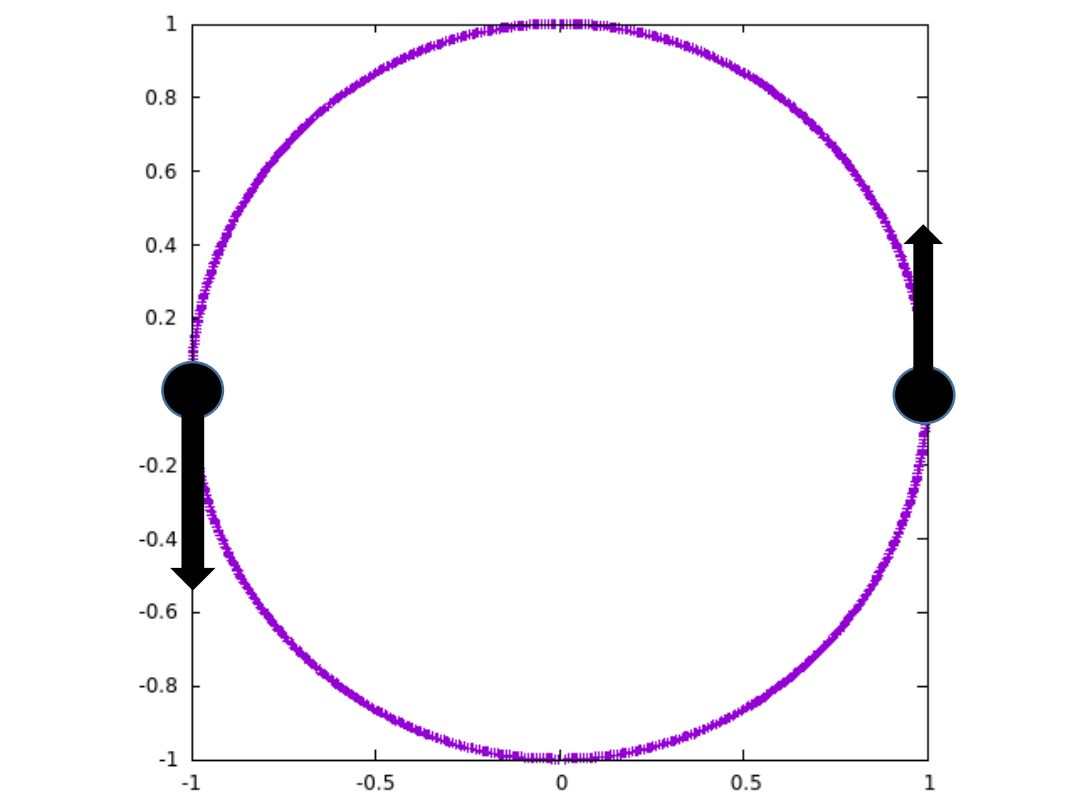
Differences in integration time: you choose a fixed REGULARIZED time step (0.01) but the corresponding physical time is evaluated through the relation $dt = R\ ds$… so the stopping time, in general, will not be EXACTLY what you specified in file parameters.txt…. Definitely not a problem !!

Simple test with B-H regularization

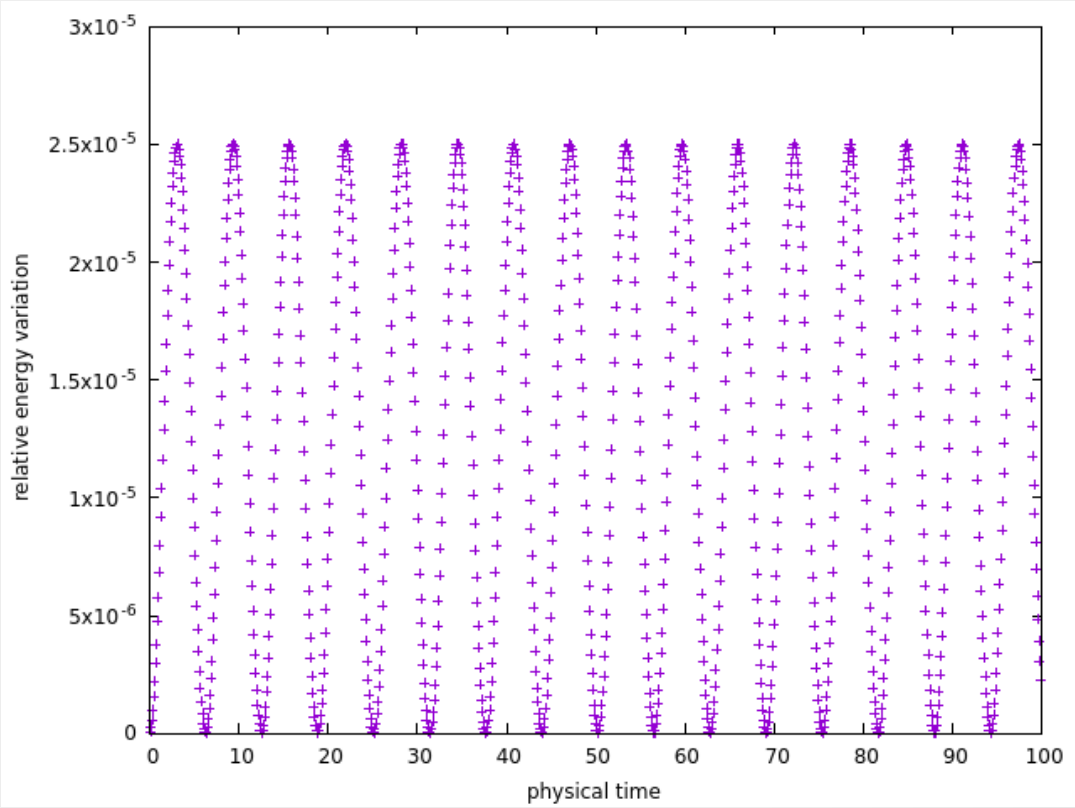
Some plots

gnuplot
plot ‘’output.dat’’ u 1:2

x-y
orbital plane



gnuplot
plot ‘’energy.dat’’ u 1:3



The orbit is **circular** (**R = 1**)

(nice !!)

→ Energy variation is **oscillating**.. No monotonic trends !! (nice !!)

→ Always **below** 2.5×10^{-5}… good value for Nbody simulations

Simple test with Hermite 4th order

cd ./nbodyclass2015/hermite0

compile the code:

./clean.sh

./make.sh

file **input.txt**

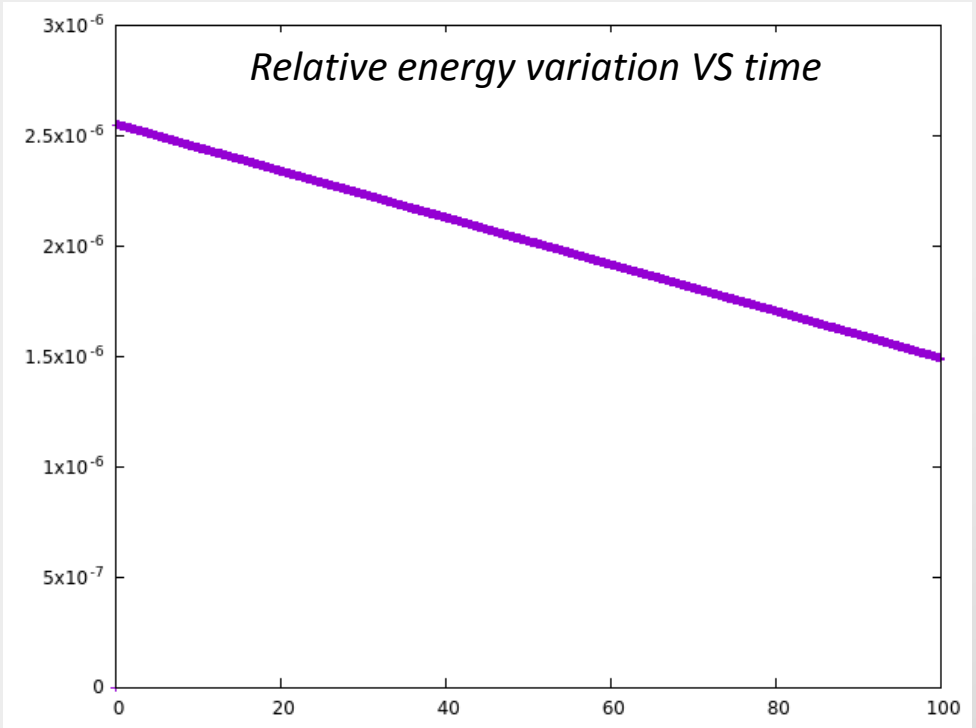
```
2   0.01   0.1   100.0    0.0
0.0 0.0 0.0 0.0 0.0 0.0 1.0
2.0 0.0 0.0 0.0 1.0 0.0 1.0


#### Format   number of particles       eta value for determining time steps     time step for snapshots     total integration time     softening
##### mass     x      y      z     vx      vy     vz
```

run using the command **./run.sh**

→ Energy variation **is not oscillating** !!!! Monotonic trend! (This is not good!!)
If we extend the numerical integration the results will be worse and worse

→ energy variation until 100 time units is **always below** $\sim 2.5 \times 10^{-6}$
*1 order of magnitude **better than the regularized code**.*



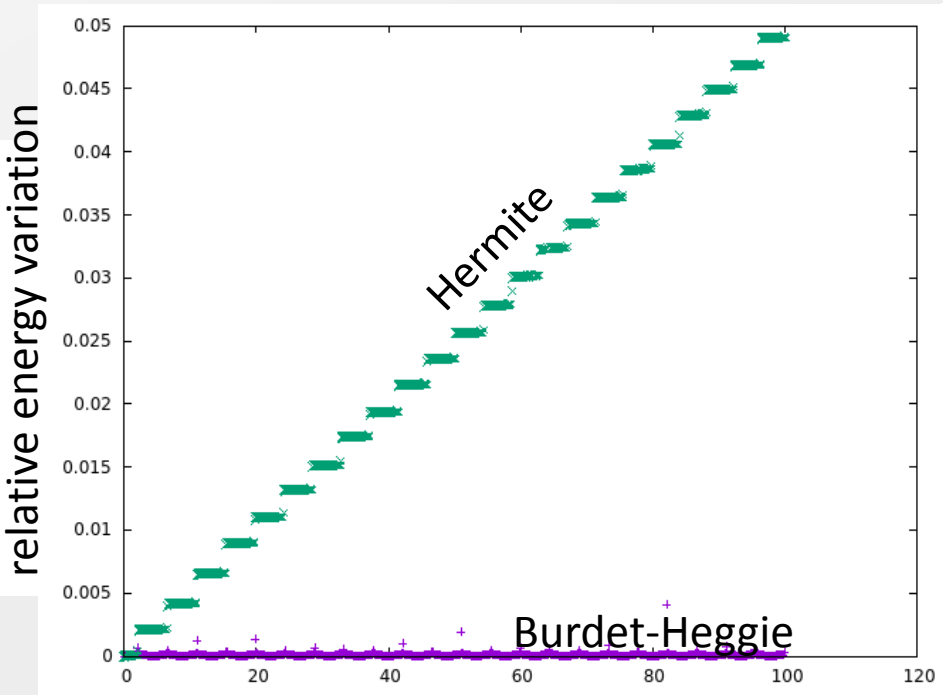
Comparison

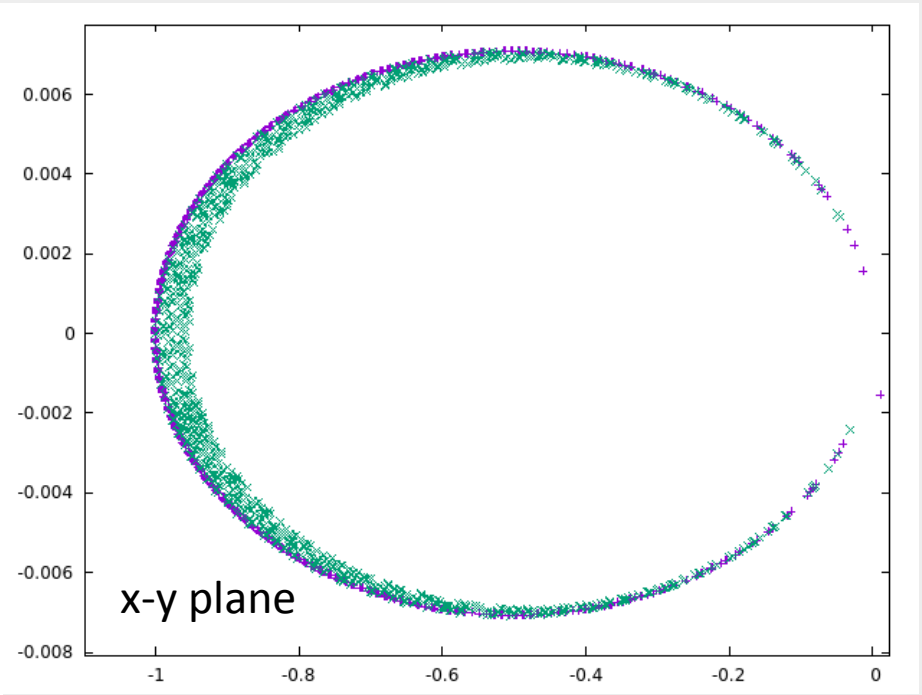
Key question #6: Hermite 4th order conserves energy better than the regularized code… Did I waste 2 hours talking about regularization?

NO!!

Key question #7: So… **what's going on?**

Well, we are using regularization to integrate a ''VERY simple'' problem: CIRCULAR orbit (no ''critical'' close encounters), equal mass stars, evolved for ~8 orbits only. Particles always have the SAME acceleration!! Let's try something harder to integrate. An orbit with eccentricity e = 0.9999 !!! (Rapidly varying position/velocity/acceleration especially around pericentre)





Hermite → $\frac{\Delta E}{E} \simeq 5 \times 10^{-2}$

(getting worse and worse)

Regularized → $\frac{\Delta E}{E} \simeq 2 \times 10^{-6}$

(oscillating)

Comparison with Mikkola's ARC

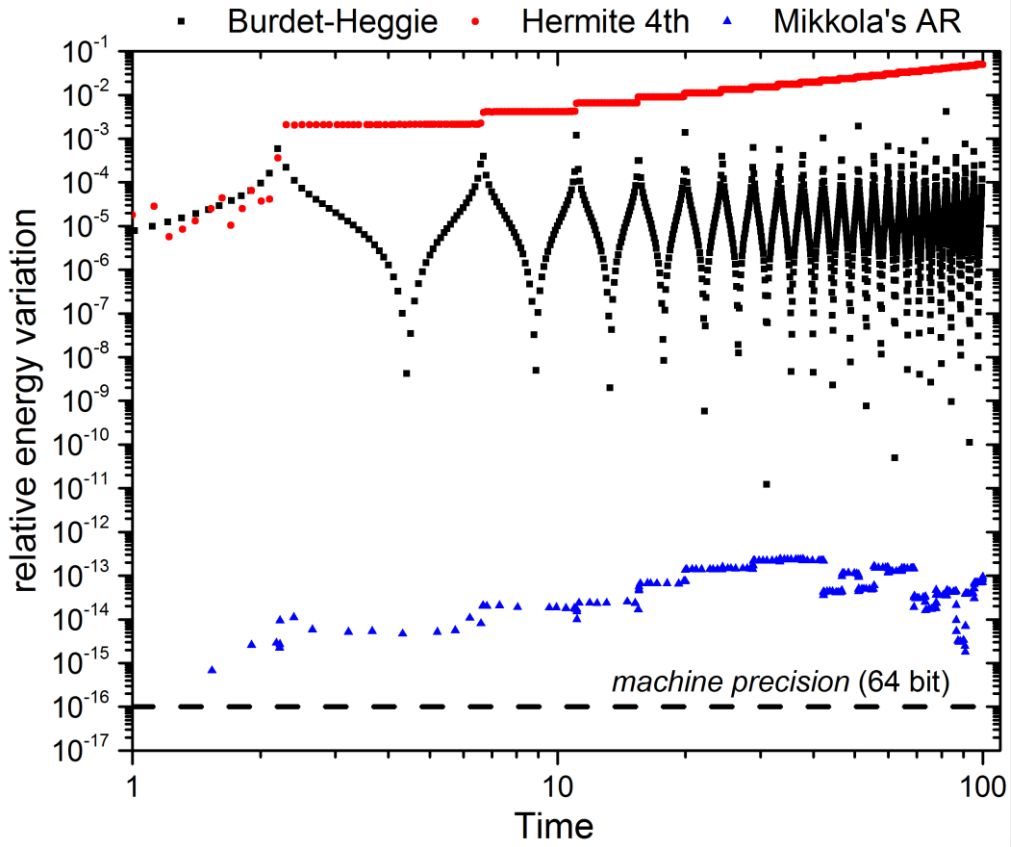
eccentric orbit (e = 0.9999)

Compile and run **./clean.sh → ./make.sh → ./run.sh**

Input.dat → initial conditions

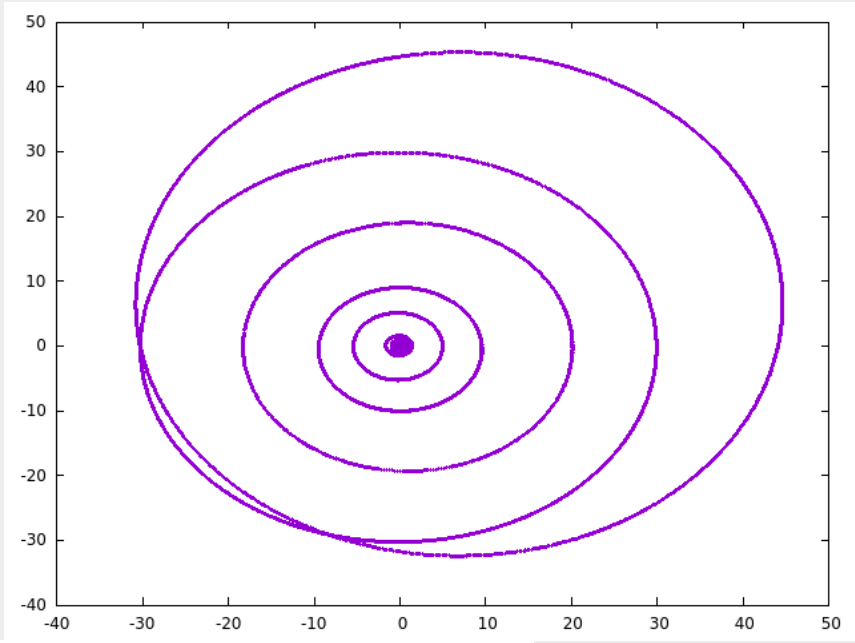
parameters.txt → simulation parameters

Just one difference→ file **energy.dat** has 2 columns ONLY
Time and energy variation



TRY IT YOURSELF !!!!

FILE: **solar_system_input.dat** → initial conditions for simulating the solar system. I took the real positions and velocities of planets+Sun at January 1902 (maybe :D) using the ephemeris I got from the NASA website… Try to run it !!



FILE: **pythagorean_3b_problem.dat** → initial conditions for 3 bodies placed at the vertices of a right triangle at rest (null initial velocity). Very common problem that it is used to «stress» N-Body code.