Elisa Bortolas

Lecture 4 – Collisional N-body simulations in Astrophysics

Computational Astrophysics Lab

University of Zurich
October 15, 2019

Animation Credits: Carl Rodriguez

Collisional Systems in Astronomy

The lifetime of the system is longer than its relaxation timescale → A system has to be treated as collisional
(in fact, we will treat a system as collisional if the time-span of the simulation is longer than the relaxation time)

Different possible equations for the **relaxation time** (i.e. the *timescale over which the velocity of a particle varies due to stellar scatterings by an amount of the order of itself*)

$$t_{\rm rlx} = 0.34 \frac{\sigma^3(r)}{G^2\, m_*\, \rho_*(r)\, \ln\Lambda}$$

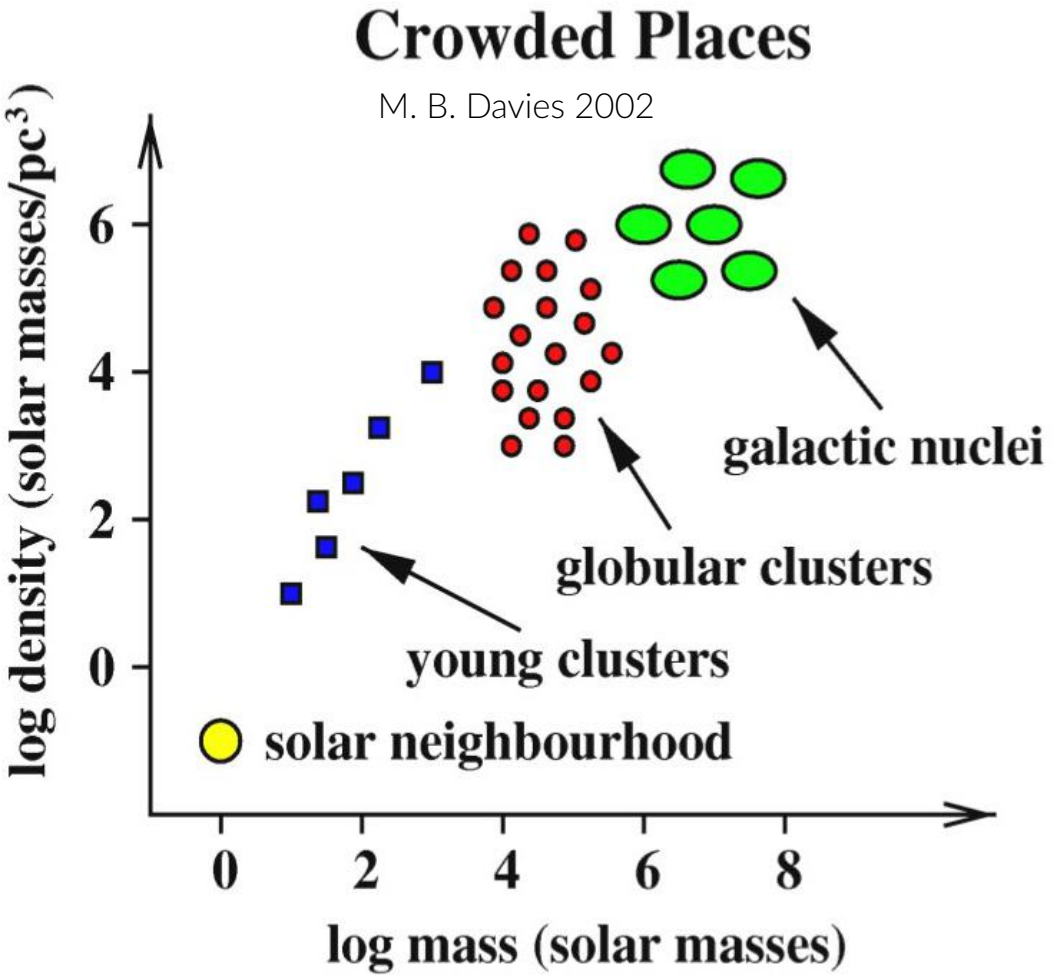
(more local)
Spitzer & Hart 1971

$$\sim 15\,{\rm Myr} \left(\frac{M_{TOT}}{10^4 M_\odot}\right)^{1/2} \left(\frac{R}{1\,{\rm pc}}\right)^{3/2} \left(\frac{1\,M_\odot}{m}\right)$$

(whole system)
Portegies Zwart 2006

$$t_{\rm relax} = \frac{N}{8 \ln N} t_{\rm cross}$$

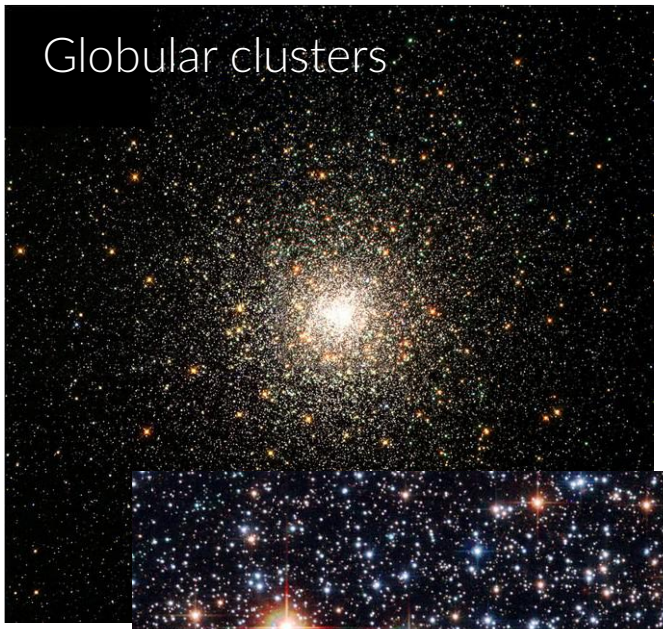
Collisional Systems in Astronomy

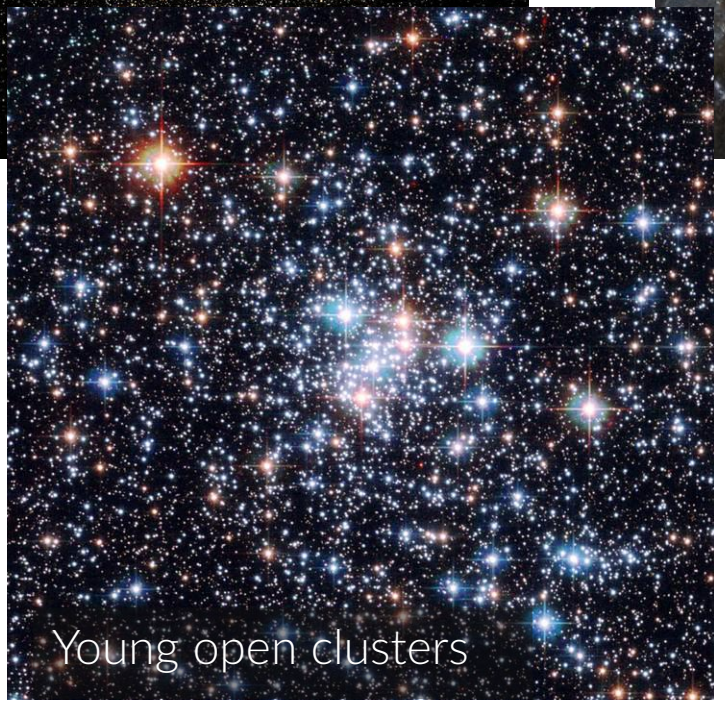


In dense systems, stars can't be assumed to move in a smooth potential, as the graininess of the system can't be disregarded

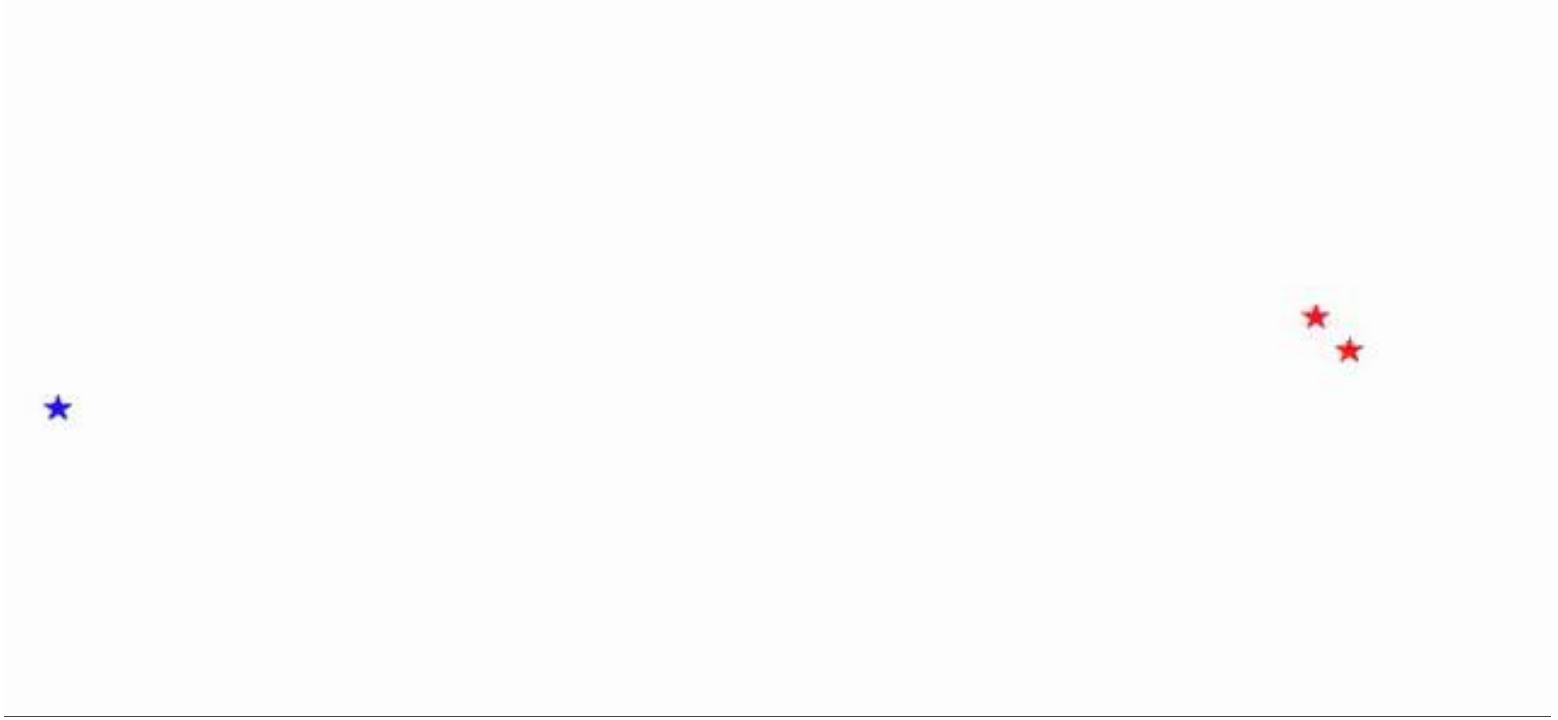
Close encounters between stars play a role in the evolution of the system and drive the evolution of the distribution function

Collisional Systems in Astronomy





Example of collisional system: 3-body system



Need for very accurate force computation and meticulous time integration, as particles do not move in a smoothly varying potential but they **experience velocity and acceleration gradients on very short timescales**

Animation Credits: Carl Rodriguez

Accurate force computation:
direct summation technique

$$\ddot{\vec{r}}_i = -G \sum_{j \neq i} m_j \frac{\vec{r}_i - \vec{r}_j}{\left|\vec{r}_i - \vec{r}_j\right|^3}$$

Direct implementation of **Newton's equations of motion** (almost always valid in galactic and planet dynamics, except very close to a massive black hole or in compact object binaries)

What is the numerical complexity?
(how many interactions to be computed?)

All pairwise forces have to be computed: N(N-1)/2

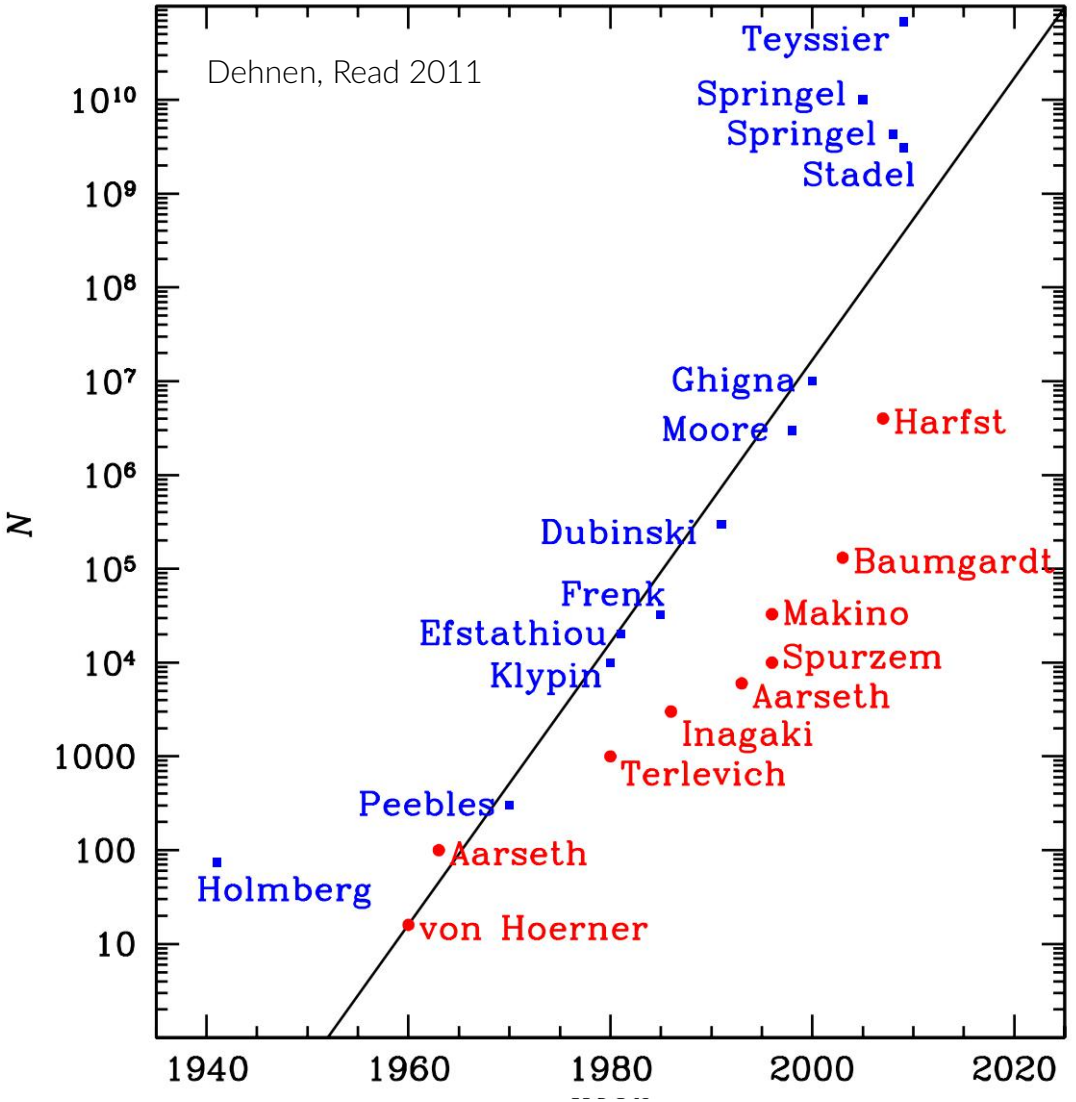
→ High computational cost, $O(N^2)$

This required when the orbit integration needs to be very accurate

Due to the large computational cost, today, it is **impossible to evolve more than a few million particles** via the direct summation technique

The increase in particle number over the past 50 years for selected collisional (red) and collisionless (blue) N-body simulations. The line shows the scaling expected from Moore's law if the costs scale $\propto$ N.
For direct summation, the scaling is in fact with N^2

Due to the expensive computation, to date direct summation is used almost solely in simulations involving ONLY PURE GRAVITY, without hydrodynamics.



Accurate time integration for the direct summation technique: the Hermite scheme

In order to integrate the orbit with high enough accuracy, we need:

i) to adopt a very small time step (a few years)
ii) to use a high order integration scheme

→ We need to conserve energy and angular momentum well during close encounters

Here we describe the 4^{th} order Hermite scheme, based on the predictor – evaluator – corrector steps

Hermite scheme: force calculation

(let's forget about softening for the moment)

Forces are computed via the direct summation method;
In fact, often one or more time derivatives of the acceleration need to be evaluated. Here for instance we need the first derivative (*jerk*)

$$\mathbf{a}_i = \sum_{j=1, j\neq i}^{N} m_j \frac{\mathbf{r}_{ij}}{x_{ij}^3};$$

$$\dot{\mathbf{a}}_i = \sum_{j=1, j\neq i}^{N} \left(m_j \frac{\mathbf{v}_{ij}}{x_{ij}^3} - 3\alpha_{ij}\mathbf{a}_{ij} \right);$$

Where:

$$\mathbf{x}_{ij} = \mathbf{x}_j - \mathbf{x}_i; \qquad \mathbf{v}_{ij} = \dot{\mathbf{x}}_j - \dot{\mathbf{x}}_i; \qquad \mathbf{a}_{ij} = \mathbf{a}_j - \mathbf{a}_i; \qquad \dot{\mathbf{a}}_{ij} = \dot{\mathbf{a}}_j - \dot{\mathbf{a}}_i;$$

$$\alpha_{ij} x_{ij}^2 = \mathbf{x}_{ij} \cdot \mathbf{v}_{ij};$$

Hermite scheme: time integration

Based on predictor – corrector – evaluator steps

Step 0:

Taylor expansion for the 4th order scheme – Δt is the integration timestep

Subscript 0: quantities at time t

Subscript 1: quantities at time t+Δt

$$\mathbf{x}_1 = \mathbf{x}_0 + \dot{\mathbf{x}}_0 \Delta t + \frac{1}{2}\mathbf{a}_0 \Delta t^2 + \frac{1}{6}\dot{\mathbf{a}}_0 \Delta t^3 + \frac{1}{24}\ddot{\mathbf{a}}_0 \Delta t^4,$$

$$\dot{\mathbf{x}}_1 = \dot{\mathbf{x}}_0 + \mathbf{a}_0 \Delta t + \frac{1}{2}\dot{\mathbf{a}}_0 \Delta t^2 + \frac{1}{6}\ddot{\mathbf{a}}_0 \Delta t^3 + \frac{1}{24}\dddot{\mathbf{a}}_0 \Delta t^4,$$

$$\begin{cases} \mathbf{a}_1 = \mathbf{a}_0 + \dot{\mathbf{a}}_0 \Delta t + \frac{1}{2}\ddot{\mathbf{a}}_0 \Delta t^2 + \frac{1}{6}\dddot{\mathbf{a}}_0 \Delta t^3, \\ \dot{\mathbf{a}}_1 = \dot{\mathbf{a}}_0 + \ddot{\mathbf{a}}_0 \Delta t + \frac{1}{2}\dddot{\mathbf{a}}_0 \Delta t^2, \end{cases}$$

The second and third derivatives of the acceleration can be eliminated via the last two equations, and substituted in the first two

Hermite scheme: time integration

$$\mathbf{x}_1 = \mathbf{x}_0 + \dot{\mathbf{x}}_0 \Delta t + \frac{1}{2}\mathbf{a}_0 \Delta t^2 + \frac{1}{6}\dot{\mathbf{a}}_0 \Delta t^3 + \frac{1}{24}\ddot{\mathbf{a}}_0 \Delta t^4,$$

$$\dot{\mathbf{x}}_1 = \dot{\mathbf{x}}_0 + \mathbf{a}_0 \Delta t + \frac{1}{2}\dot{\mathbf{a}}_0 \Delta t^2 + \frac{1}{6}\ddot{\mathbf{a}}_0 \Delta t^3 + \frac{1}{24}\dddot{\mathbf{a}}_0 \Delta t^4,$$

$$\mathbf{a}_1 = \mathbf{a}_0 + \dot{\mathbf{a}}_0 \Delta t + \frac{1}{2}\ddot{\mathbf{a}}_0 \Delta t^2 + \frac{1}{6}\dddot{\mathbf{a}}_0 \Delta t^3,$$

$$\dot{\mathbf{a}}_1 = \dot{\mathbf{a}}_0 + \ddot{\mathbf{a}}_0 \Delta t + \frac{1}{2}\dddot{\mathbf{a}}_0 \Delta t^2,$$

$$\dot{\mathbf{x}}_1 = \dot{\mathbf{x}}_0 + \frac{1}{2}(\mathbf{a}_0 + \mathbf{a}_1)\Delta t + \frac{1}{12}(\dot{\mathbf{a}}_0 - \dot{\mathbf{a}}_1)\Delta t^2,$$

$$\mathbf{x}_1 = \mathbf{x}_0 + \frac{1}{2}(\dot{\mathbf{x}}_0 + \dot{\mathbf{x}}_1)\Delta t + \frac{1}{12}(\mathbf{a}_0 - \mathbf{a}_1)\Delta t^2.$$

It means that:
(1) we have a 4th order scheme with only 2nd order terms
(2) it is an implicit scheme: in order to compute positions and velocities at time 1, we need to know velocities, accelerations and jerks in the future
→ need for the predictor step

Hermite 4th order

$$\dot{\mathbf{x}}_1 = \dot{\mathbf{x}}_0 + \tfrac{1}{2}(\mathbf{a}_0 + \mathbf{a}_1)\Delta t + \tfrac{1}{12}(\dot{\mathbf{a}}_0 - \dot{\mathbf{a}}_1)\Delta t^2,$$
$$\mathbf{x}_1 = \mathbf{x}_0 + \tfrac{1}{2}(\dot{\mathbf{x}}_0 + \dot{\mathbf{x}}_1)\Delta t + \tfrac{1}{12}(\mathbf{a}_0 - \mathbf{a}_1)\Delta t^2.$$

Comparison with Leapfrog kick – drift -kick

$$\boldsymbol{x}_1 = \boldsymbol{x}_0 + \dot{\boldsymbol{x}}_0\Delta t + \frac{1}{2}\boldsymbol{a}_0\Delta t^2,$$
$$\dot{\boldsymbol{x}}_1 = \dot{\boldsymbol{x}}_0 + \frac{1}{2}(\boldsymbol{a}_0 + \boldsymbol{a}_1)\Delta t.$$

Predictor step

$$\mathbf{x}_p = \mathbf{x}_0 + \dot{\mathbf{x}}_0 \Delta t + \frac{1}{2}\mathbf{a}_0 \Delta t^2 + \frac{1}{6}\dot{\mathbf{a}}_0 \Delta t^3,$$
$$\dot{\mathbf{x}}_p = \dot{\mathbf{x}}_0 + \mathbf{a}_0 \Delta t + \frac{1}{2}\dot{\mathbf{a}}_0 \Delta t^2;$$

Newton formula is adopted to compute the current quantities at time t ('0' quantities)
The predicted (p) positions and velocities are found

Evaluator step

The predicted quantities x_p, $\dot{x}_p$ are used to evaluate the future (time t+Δt, subscript '1') accelerations and jerks from Newton's formula

$$\mathbf{a}_i = \sum_{j=1, j\neq i}^{N} m_j \frac{\mathbf{r}_{ij}}{x_{ij}^3};$$

$$\dot{\mathbf{a}}_i = \sum_{j=1, j\neq i}^{N} \left(m_j \frac{\mathbf{v}_{ij}}{x_{ij}^3} - 3\alpha_{ij}\mathbf{a}_{ij} \right);$$

Corrector step

The newly evaluated accelerations and jerks (from predicted quantities) are used inside the previous Taylor expansion to compute the final positions and velocities

$$\dot{\mathbf{x}}_1 = \dot{\mathbf{x}}_0 + \frac{1}{2}(\mathbf{a}_0 + \mathbf{a}_1)\Delta t + \frac{1}{12}(\dot{\mathbf{a}}_0 - \dot{\mathbf{a}}_1)\Delta t^2,$$
$$\mathbf{x}_1 = \mathbf{x}_0 + \frac{1}{2}(\dot{\mathbf{x}}_0 + \dot{\mathbf{x}}_1)\Delta t + \frac{1}{12}(\mathbf{a}_0 - \mathbf{a}_1)\Delta t^2.$$

Note that if we were computing the corrected position first, and then the corrected velocity, the result would be only 3rd order in position.
The trick here is **first to compute the corrected velocity, and then to use it for evaluating the corrected position**

Choice of timestep

What is the ideal timestep Δt?

It is easy to guess that different particles in our simulations would have different ideal choice of time-step, depending on how fast their velocities and accelerations are changing

Ideas from dimensional analysis?

(hint for easy guess: softening is still around)

Choice of timestep

What is the ideal timestep Δt?

It is easy to guess that different particles in our simulations would have different ideal choice of time-step, depending on how fast their velocities and accelerations are changing

Possible criteria:

$$\Delta t_i = \eta \sqrt{\frac{\varepsilon}{|\mathbf{a}_i|}} \qquad \Delta t_i = \eta \left(\frac{|\mathbf{a}_i|}{|\mathbf{a}_i^{(p)}|} \right)^{\frac{1}{p}} .$$

ε is the softening
η is a accuracy parameter ~0.1-0.2;
(p) is the p-th derivative of the acceleration

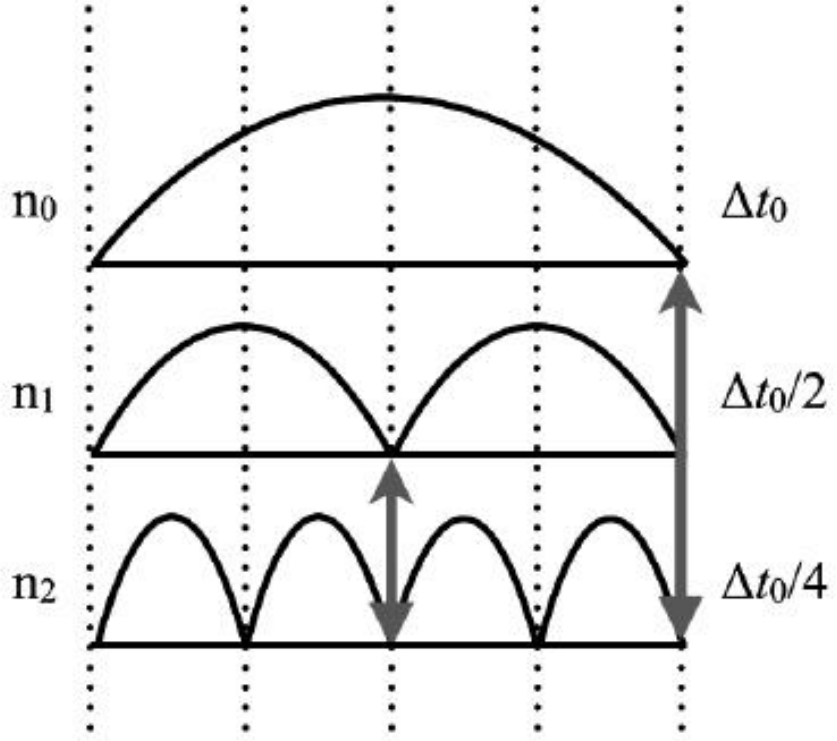
Best performing criterion so far:

$$\Delta t_i = \left(\eta \frac{|\mathbf{a}_i||\ddot{\mathbf{a}}_i| + |\dot{\mathbf{a}}_i|^2}{|\dot{\mathbf{a}}_i||\dddot{\mathbf{a}}_i| + |\ddot{\mathbf{a}}_i|^2} \right)^{\frac{1}{2}} .$$

This timestep choice owes its efficiency to the fact that it shrinks the timestep if the second or third acceleration derivatives are large compared to the smaller derivatives.

Block time steps

If all particles were having a different timestep, the computational cost would be insanely expensive: the solution is to use **BLOCK timesteps**



$$\Delta t_n = 2^{-n} \Delta t_0$$

The timesteps are organized into a **hierarchical structure**, and each particle is assigned to a different block in the hierarchy via the previous criteria.

The timestep assigned to a given particle should be the one in the hierarchy just below (smaller) than the one obtained with the previous criteria

Note that: **quantities are predicted for all particles, but corrected only for those particles that need to be synchronized** in the predictor-corrector-evaluator scheme

Softening & Softening Kernel

$$\Phi(\mathbf{r}_i) = -G \sum_{j=1, j\neq i}^{j=N} S(r_{ij}, \varepsilon) m_i m_j. \qquad \mathbf{F}_i = -G m_i \sum_{j=1,\ j\neq i}^{j=N} S_F(r_{ij}, \varepsilon) m_j \frac{\mathbf{r}_i - \mathbf{r}_j}{r_{ij}},$$

Plummer softening Kernel (S_F is the derivative of S with respect to r_{ij})

$$S(r_{ij}, \varepsilon) = -\frac{1}{\sqrt{r_{ij}^2 + \varepsilon^2}}.$$

Surprise! Softening is still around (ε)

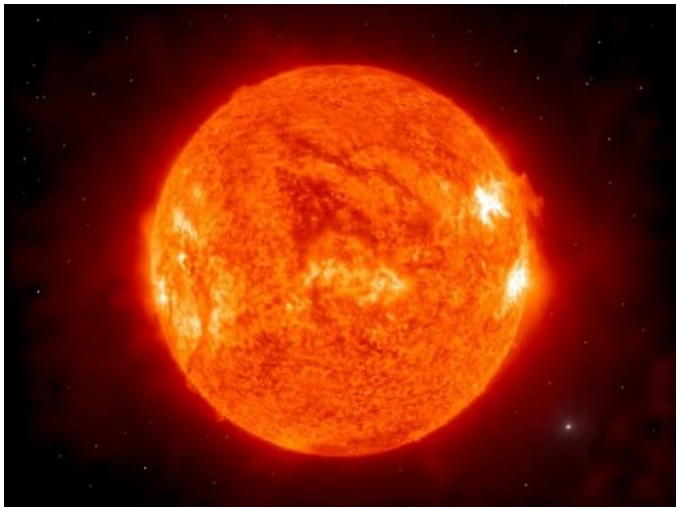
$$\vec{a}_i = -G \sum_{j \neq i} \frac{m_j \vec{r}_{ij}}{\left(r_{ij}^2 + \varepsilon^2\right)^{3/2}}$$

$$\frac{d\vec{a}_i}{dt} = -G \sum_{j \neq i} m_j \left[\frac{\vec{v}_{ij}}{\left(r_{ij}^2 + \varepsilon^2\right)^{3/2}} - \frac{3(\vec{v}_{ij} \cdot \vec{r}_{ij})\vec{r}_{ij}}{\left(r_{ij}^2 + \varepsilon^2\right)^{5/2}} \right]$$

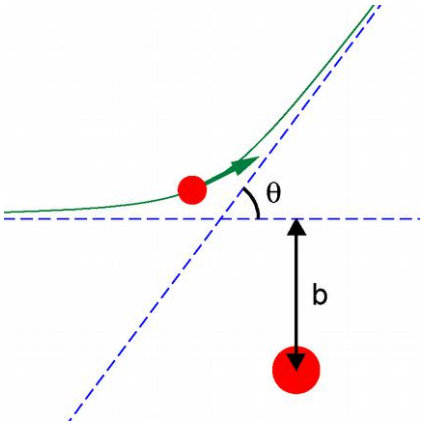
Where: $\vec{r}_{ij} = \vec{x}_i - \vec{x}_j; \quad r_{ij}^2 = (\vec{x}_i - \vec{x}_j)^2$

$\vec{v}_{ij} = \vec{v}_i - \vec{v}_j$

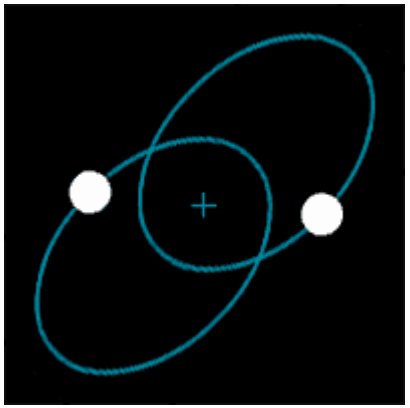
Can you guess why?



1. Physical sizes of stars

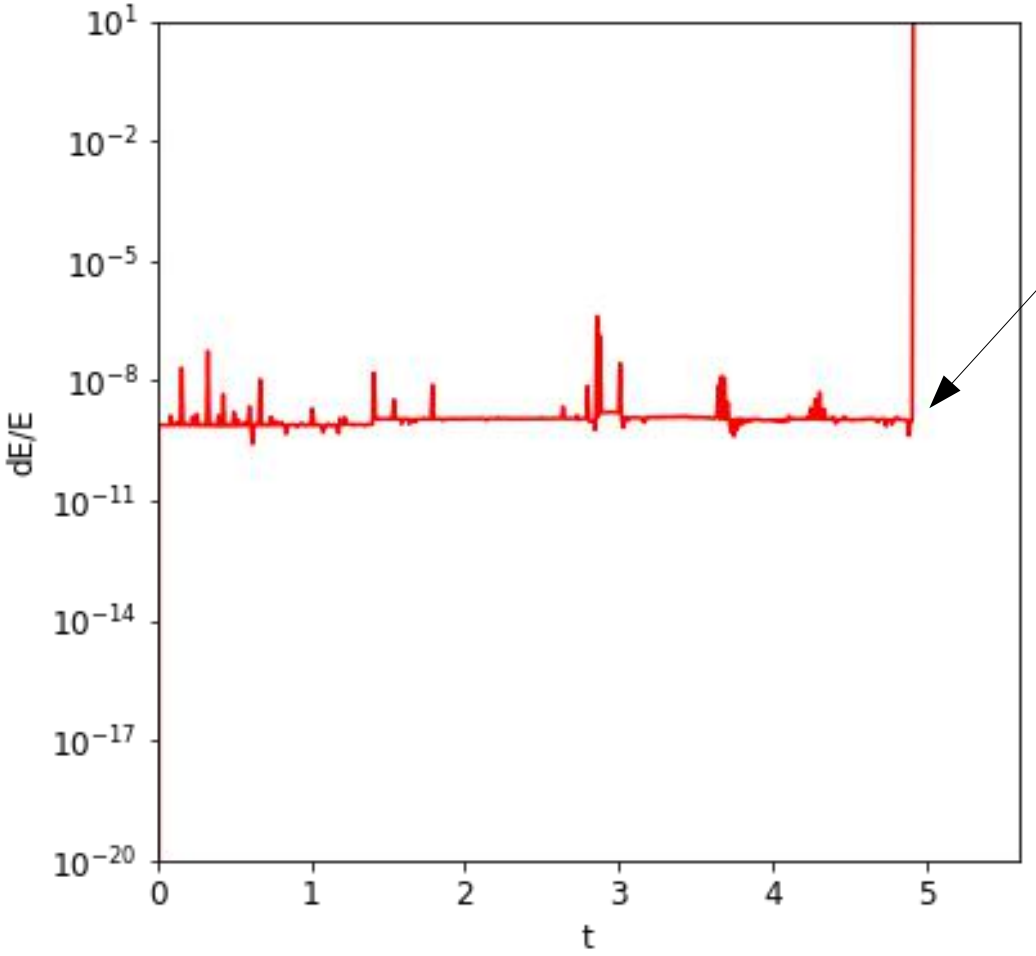


2a. Very close interactions:
insanely small timestep would be needed



2b. Formation of tight binaries:
the simulation would never end!

How to check the accuracy of the integration:
Energy (and Angular Momentum) Conservation



Spurious scattering! (perhaps too small softening?)

$|dE/E| = |[E_i - E(t)]/E_i|$ should be below ~10^-4; E_i is the energy at the beginning of the integration

Note that energy computation should account for softening:

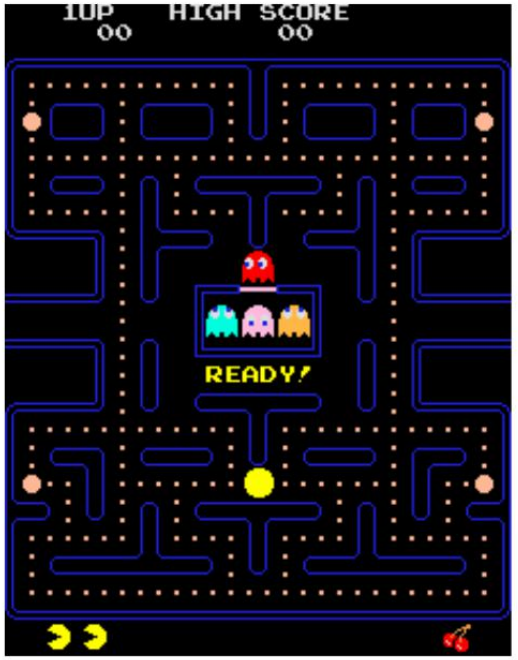
$$E = \sum_{i=1}^{N} m_i \left(\frac{v_i^2}{2} - \sum_{j<i} \frac{G m_j}{\sqrt{(\mathbf{r}_j - \mathbf{r}_i)^2 + \epsilon^2}} \right)$$

Architectures for direct summation: Graphic Processors (GPUs)

Wikipedia's definition: *specialized electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display*

Born for applications that need FAST and HEAVY GRAPHICS: VIDEO GAMES

BEFORE GPU



AFTER GPU

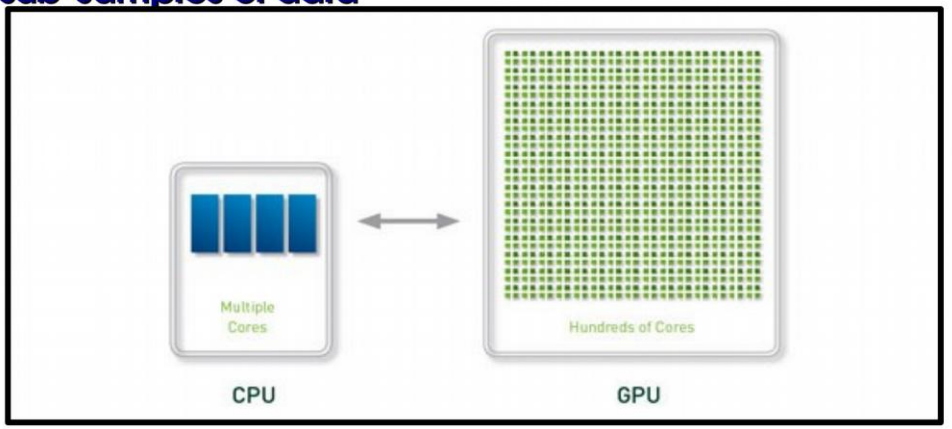
Credit: M. Mapelli

Architectures for direct summation: Graphic Processors (GPUs)

SIMD/SIMT TECHNIQUE: SINGLE INSTRUCTION MULTIPLE DATA/THREADS

many processing units perform the same series of operations

on different sub-samples of data



Even current CPUs are multiple CORES (i.e. can be multi-threading)

but the number of independent cores in GPUs is ~100 times larger!

Why are they particularly good for direct summation?
BECAUSE THEY DO A SINGLE OPERATION (acceleration and jerk calculation)
on MANY PAIRS of PARTICLES EACH INTER-PARTICLE FORCE BETWEEN A PAIR IS INDEPENDENT OF THE OTHER PAIRS

$$\mathbf{a}_i = \sum_{j=1, j\neq i}^{N} m_j \frac{\mathbf{r}_{ij}}{x_{ij}^3};$$

$$\dot{\mathbf{a}}_i = \sum_{j=1, j\neq i}^{N} \left(m_j \frac{\mathbf{v}_{ij}}{x_{ij}^3} - 3\alpha_{ij}\mathbf{a}_{ij} \right);$$

Credit: M. Mapelli

Summary

Direct summation is the technique to be used for collisional systems (t_{rel}< lifetime) due to high accuracy – single encounters can be well resolved

Hermite integration (especially 4^{th} order) is the most used technique as it is high order and it allows block timesteps (these are typically computed via the Aarseth criterion)

Softening is still needed to avoid spurious scatterings (if the timestep is too large compared to the change in acceleration or velocity), to account for the physical size of stars, to avoid the formation of tight binaries

Graphic processors are used to integrate via the direct summation techniques: current state is a few million particles at most

Caveats

If $N_{simulated} << N_{real} \rightarrow$ spurious two body relaxation, be careful!

Even when simulating real systems, tight binaries or stellar collisions cannot be accounted for easily

Evolution in the close vicinity to a supermassive black hole or for very tight binaries would require a regularization scheme (change of coordinates to avoid the singularity at r = 0, VERY EXPENSIVE and non parallel); stellar evolution should also ideally be included

In the presence of compact objects the general relativistic corrections to the Newtonian dynamics should be accounted for (inclusion of Post-Newtonian terms)

Bibliography

This lecture is mainly based on:
https://web.oapd.inaf.it/mapelli/lectures.html (N-body techniques, see year 2018 Lecture 3 – or year 2015 Lecture2)

Further reading:
https://arxiv.org/abs/1105.1082 (Dehnen Read 2011, advances in collisionless and collisional simulations)

https://drive.google.com/file/d/0Bx7KflRPRjKbZTRkc19yeXdLc1E/view (chapter 3, Master thesis)

https://ui.adsabs.harvard.edu/abs/2007NewA...12..357H/abstract (Harfst+ 2007, architectures for direct summation)

https://www.cambridge.org/core/books/gravitational-millionbody-problem/AFA2C2F4821B5FEF991A22470E889C22
(nice introduction on collisional dynamics from a different point of view)

http://www.artcompsci.org/kali/vol/vision/title.html Dealing with dense stellar systems, Hut & Makino

Bibliography

This lecture is mainly based on:
https://web.oapd.inaf.it/mapelli/lectures.html (N-body techniques, see year 2018 Lecture 3 – or year 2015 Lecture2)

Further reading:
https://arxiv.org/abs/1105.1082 (Dehnen Read 2011, advances in collisionless and collisional simulations)

https://drive.google.com/file/d/0Bx7KflRPRjKbZTRkc19yeXdLc1E/view (chapter 3, Master thesis)

https://ui.adsabs.harvard.edu/abs/2007NewA...12..357H/abstract (Harfst+ 2007, architectures for direct summation)

https://www.cambridge.org/core/books/gravitational-millionbody-problem/AFA2C2F4821B5FEF991A22470E889C22
(nice introduction on collisional dynamics from a different point of view)

http://www.artcompsci.org/kali/vol/vision/title.html Dealing with dense stellar systems, Hut & Makino