STELLAR DYNAMICS

D.C. Heggie
University of Edinburgh
d.c.heggie@ed.ac.uk

Outline

1. A primer in stellar dynamics
2. *N*-body codes

The N-body Problem

- N point masses (good while separation of two stars is much less than the sum of their radii)
- classical gravitation and equations of motion (good except close to horizon of black hole, or close binaries emitting gravitational waves)

Equations of motion

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

or

$$\begin{cases} \dot{\mathbf{r}}_i = \mathbf{v}_i \\ \dot{\mathbf{v}}_i = -G \sum_{j=1,\neq i}^{N} m_j \frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|^3} \end{cases}$$

Cold collapse

Initial conditions:

- All velocities 0
- Particles distributed uniformly in sphere

The crossing time and virial equilibrium

- The system "quickly" reaches a "steady state"
- The steady state is in "virial equilbrium", when the virial equation is approximately satisfied:

$$2T + V = 0$$

where

$$T = \frac{1}{2}\sum_{i=1}^{N} m_i \mathbf{v}_i^2 \quad \text{(Kinetic Energy)}$$

$$V = -\frac{G}{2}\sum_{i=1}^{N}\sum_{j=1,\neq i}^{N} \frac{m_j m_i}{|\mathbf{r}_i - \mathbf{r}_j|} \quad \text{(Potential Energy)}$$

Mass, length and time scales

- Total mass

$$M = \sum_{i=1}^{N} m_i$$

- Characterise system size by "virial radius" R defined by

$$V = -\frac{GM^2}{2R}, \quad \text{where } M \text{ is total mass}$$

- Characterise speeds by (mass weighted) mean square speed

$$v^2 = \frac{2T}{M}$$

- Define time scale

$$t_{cr} = \frac{2R}{v} \quad (\text{"Crossing time"})$$

N-body Units

A conventional system of units in which

$$\begin{aligned} G &= 1 \\ M &= 1 \\ R &= 1 \end{aligned}$$

Example

Suppose a star cluster has $M = 10^5 M_\odot, R = 5\text{pc}$. To convert a velocity from the *N*-body code to km/s, multiply by $\sqrt{\frac{GM}{R}}$, where G is expressed in the same units (i.e. km/s, $M_\odot$, pc), i.e. $G \simeq 0.043$.

Significance of the crossing time

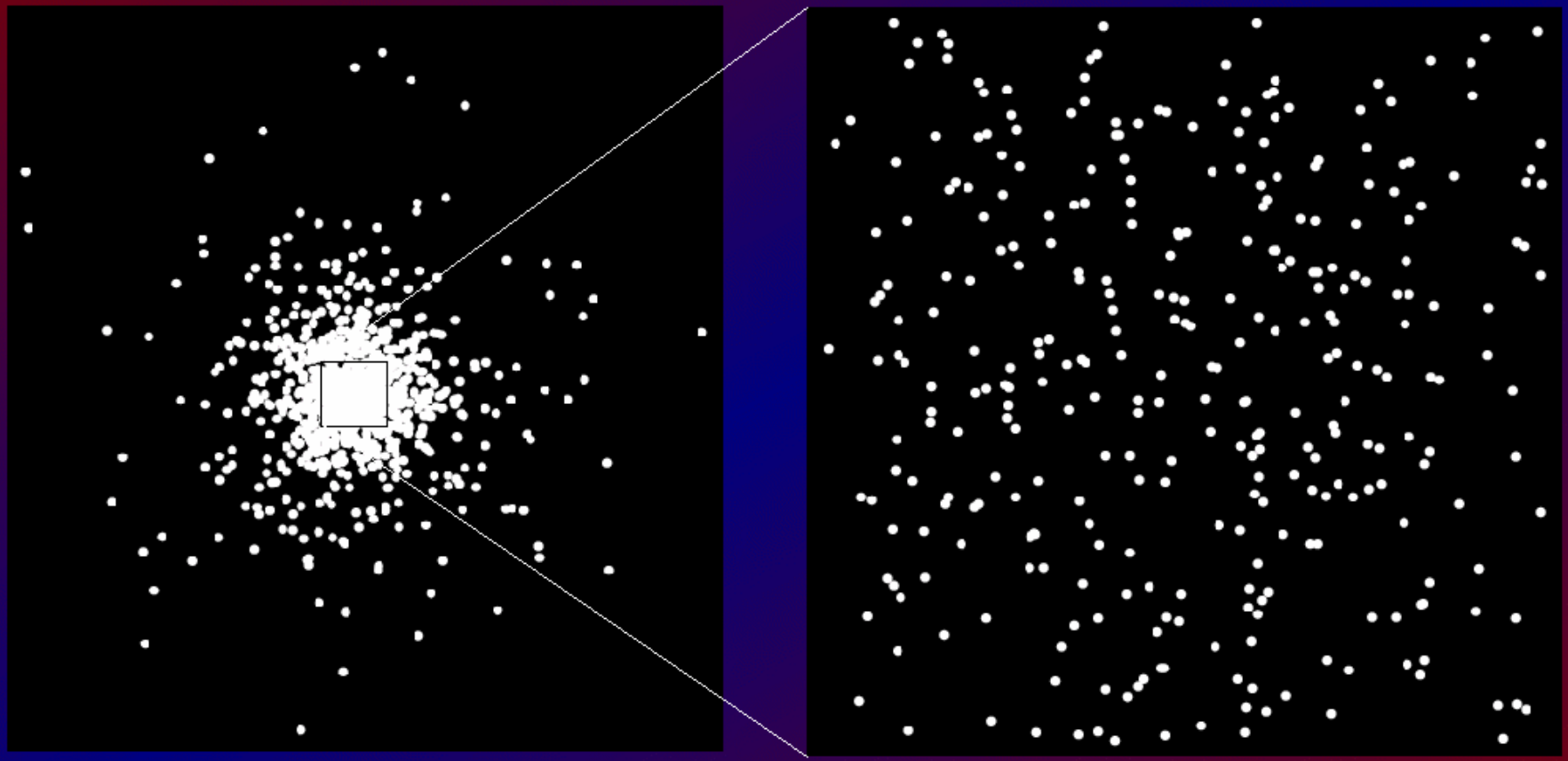
- Time scale of cold collapse
- Time scale of approach to virial equilibrium
- Time scale of orbital motions in virial equilibrium

Plummer's model

- A particular model of a system in virial (and dynamic) equilibrium
- Convenient analytical distributions

YITP

Core collapse: Two simulations



Entire system
13 seconds to t = 3.3

Central area
13 seconds to t = 330

The initial conditions (Plummer's model)

Lessons from the simulations

Evolution on two time scales:

- orbital motions (crossing time scale)
- much slower evolution of the statistical distribution

Two-body relaxation

- Local definition $t_r = \dfrac{0.065 v^3}{\rho m G^2 \ln \gamma N}$ where
  - v is the velocity dispersion (root mean square velocity)
  - ρ is the space (mass-)density
  - m is the particle mass
  - γ is a constant (about 0.11 for equal masses)
  - N is number of particles
- Global definition: half-mass relaxation time

  $t_{rh} = 0.138 \dfrac{N^{1/2} r_h^{3/2}}{m^{1/2} G^{1/2} \ln(\gamma N)}$, where
  - r_h is the *half-mass radius* (containing the innermost half of the system); comparable with the virial radius

Significance of the relaxation time

- Time scale of core collapse
- Time scale of escape (actually several/many t_r)
- Time scale of *mass segregation* (if there is a distribution of masses, the heavier particles sink to the centre on a time scale which is a fraction of t_r)

- *Collisional* stellar dynamics deals with phenomena on time scales of a few t_{rh} (open and globular star clusters; some galactic nuclei)
- *Collisionless* stellar dynamics deals with phenomena on time scales much less than t_{rh} (spiral structure, galaxy collisions (!), ...)

Post-collapse evolution

Depends on boundary conditions:

- "isolated" system: binaries form in the core, liberating energy, which expands the system on the time scale t_{rh} (by a feedback mechanism). As r_h expands, t_{rh} increases. The system *very* slowly loses mass
- "tidally limited" systems: stars escape (roughly speaking) at a tidal radius r_t, where external forces become dominant. Mass is lost on time scale t_{rh}; r_t contracts, r_h contracts (eventually). System dissolves in few t_{rh}.

N-body codes

Example: Euler method
Equations of motion:

$$\begin{aligned} \dot{\mathbf{r}}_i &= \mathbf{v}_i \\ \dot{\mathbf{v}}_i &= \mathbf{a}_i = -\sum_{j=1, j\neq i}^{N} G m_j \frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|^3} \end{aligned}$$

Algorithm:

$$\begin{aligned} \mathbf{r}_i(t + \Delta t) &= \mathbf{r}_i(t) + \Delta t\ \mathbf{v}_i(t) \\ \mathbf{v}_i(t + \Delta t) &= \mathbf{v}_i(t) + \Delta t\ \mathbf{a}_i(t) \end{aligned}$$

Note: singularity where $\mathbf{r}_i = \mathbf{r}_j$ (collision between any pair of particles)

Time step limited by $\Delta t < \eta \min_{i,j} \dfrac{|\mathbf{r}_i - \mathbf{r}_j|}{|\mathbf{v}_i - \mathbf{v}_j|}$

Choices of Time Step

- *Fixed* time step - too short, can't be predicted - impractical
- Variable *shared* time step - forces all particles to take same Δt - impractical. MUSE example: Hermite0
- Variable *individual* time step - near-optimal, but requires extrapolation. MUSE example: nbody1h
- Block time steps ($\Delta t = 2^{-k}, k = 0, 1, 2, \ldots$) - shares extrapolation

Conventional choices

Time step: a generalisation of $\eta \min_{i,j} \dfrac{|\mathbf{r}_i - \mathbf{r}_j|}{|\mathbf{v}_i - \mathbf{v}_j|}$

Particle advance: a generalisation of Euler called *Hermite*
First step:

$$
\begin{aligned}
\text{Euler:}\mathbf{r}_i &:= \mathbf{r}_i + \mathbf{v}_i \Delta t \\
\mathbf{v}_i &:= \mathbf{v}_i + \mathbf{a}_i \Delta t \\
\text{Hermite:}\mathbf{r}_i &:= \mathbf{r}_i + \mathbf{v}_i \Delta t + \frac{1}{2}\mathbf{a}_i \Delta t^2 + \frac{1}{6}\dot{\mathbf{a}}_i \Delta t^3 \\
\mathbf{v}_i &:= \mathbf{v}_i + \mathbf{a}_i \Delta t + \frac{1}{2}\dot{\mathbf{a}}_i \Delta t^2
\end{aligned}
$$

followed by a corrector involving values of $\mathbf{a}_i, \dot{\mathbf{a}}_i$ at the *end* of the time step (Hermite only).

Basic structure of an N-body code

1. Initialisation of $\mathbf{r}_i, \mathbf{v}_i, tnext_i$ (update time), $\mathbf{a}_i, \dot{\mathbf{a}}_i$, all i.
2. Choose i minimising $tnext_i$
3. Extrapolate all $\mathbf{r}_j$ to $tnext_i$
4. Compute new $\mathbf{a}_i, \dot{\mathbf{a}}_i$
5. Correct new $\mathbf{r}_i$, compute new $\mathbf{v}_i$ (Hermite integrator)
6. Compute new $tnext_i$
7. Repeat from step 2

Notes

- this does not include block time steps
- this is the basic structure of NBODY1 (Aarseth)

Accelerating force calculation: software

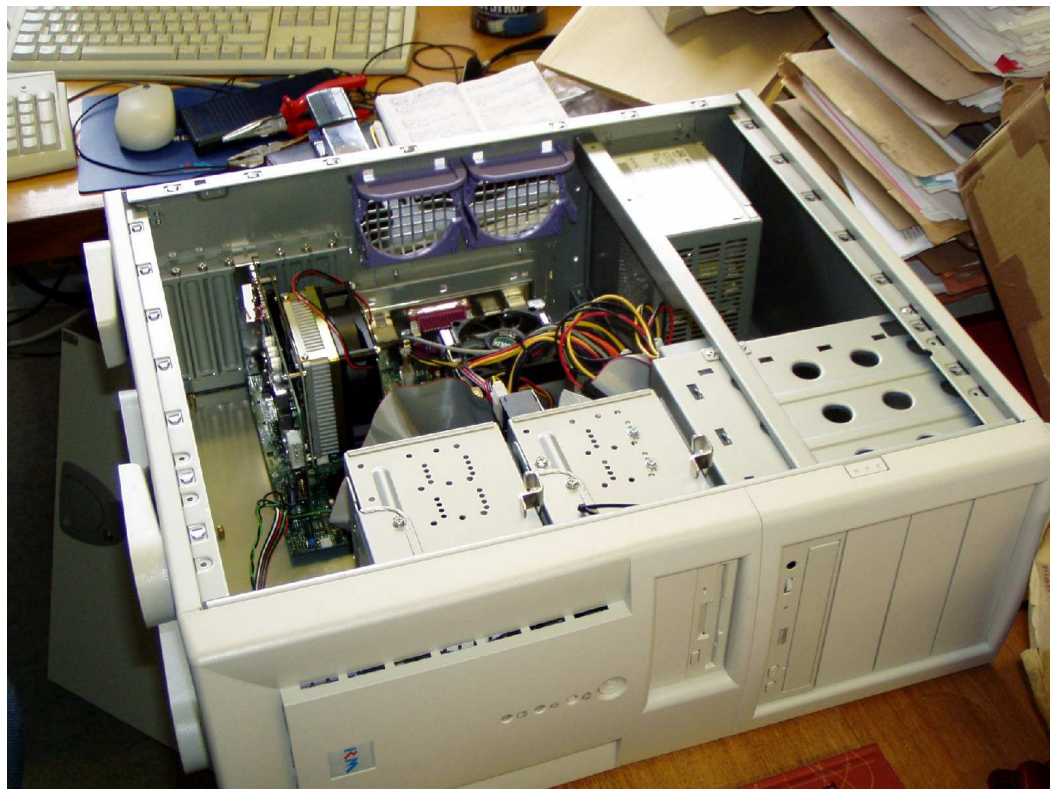
Neighbour Scheme (Ahmad-Cohen)

- For each particle i, keep a list of its near neighbours
- Update the neighbour force frequently, the non-neighbour force less frequently ("regular" and "irregular" forces)

Tree code: little used in collisional simulations. MUSE example: BHTree

Accelerating force calculation: hardware

- GRAPE
  - Entry level version costs a few thousand euros



  - Speed-up factor ~ 100
- Parallel computers and clusters: NBODY6++ (Spurzem)
- Video cards (Portegies Zwart)

Close encounters and binaries. I. Offset "regularisation"

Suppose particles i, j form a bound pair, or experience a close encounter. Use offset variables $\mathbf{r}, \mathbf{R}$ defined as

$$\begin{aligned} \mathbf{R} &= \frac{m_i \mathbf{r}_i + m_j \mathbf{r}_j}{m_i + m_j} \\ \mathbf{r} &= \mathbf{r}_i - \mathbf{r}_j, \end{aligned}$$

and write equations of motion in terms of $\mathbf{r}, \mathbf{R}$: e.g.

$$\ddot{\mathbf{r}} = -G(m_i + m_j)\frac{\mathbf{r}}{|\mathbf{r}|^3} + \mathbf{a}'_i - \mathbf{a}'_j,$$

where $'$ means we omit force due to i, j.
Advantage: avoids rounding error in repeated calculation of $\mathbf{r}_i - \mathbf{r}_j$.

Close encounters and binaries. II. KS regularisation

- Singularity in

$$\ddot{\mathbf{r}} = -G(m_i + m_j)\frac{\mathbf{r}}{|\mathbf{r}|^3} + \mathbf{a}'_i - \mathbf{a}'_j$$

  requires small time steps for close and/or eccentric binaries.
- KS regularisation is subtle change of variables which removes the singularity.
- still requires short time step for close binary
- freeze unperturbed binaries

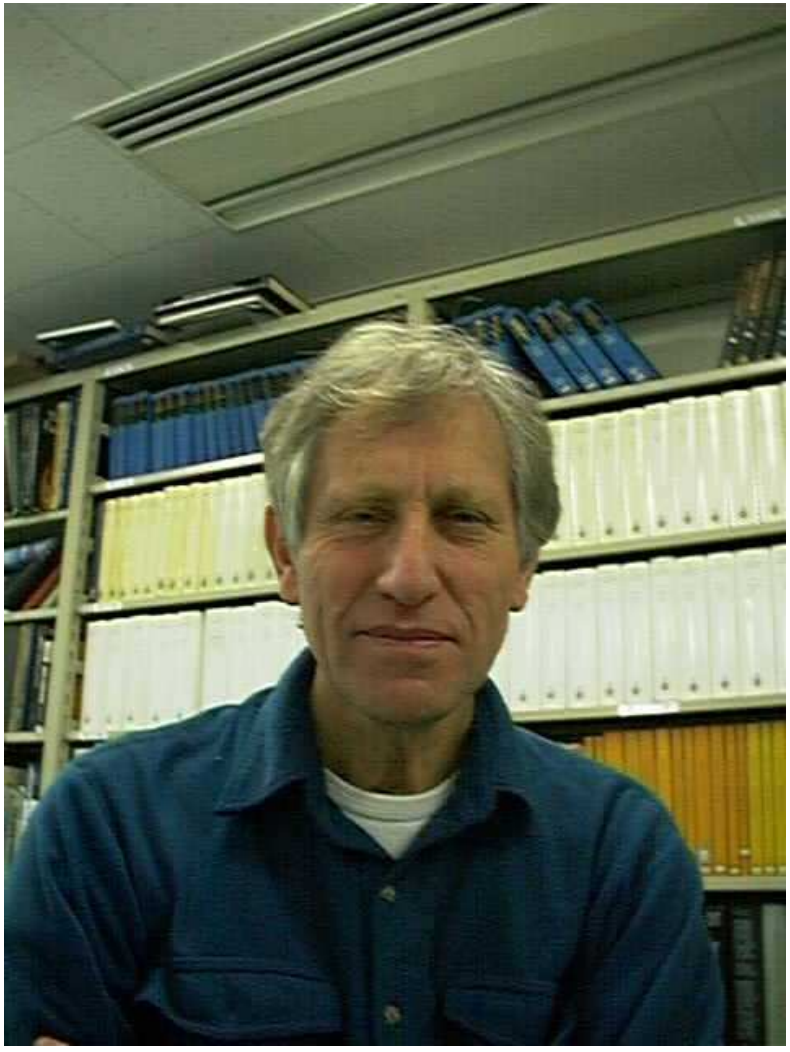
Higher-order subsystems: triples, quadruples, etc

- hierarchical triples are binaries constantly perturbed by third body: "slow-down" treatment follows secular perturbations with (much) larger time step
- non-hierarchical triples, quadruples: *chain regularisation*, a generalisation of offset and KS regularisation; there are specilisations to triples and quadruples

Flow control

Each integration step may involve any of the following possibilities (Aarseth, NBODY6):

1. Standard integration
2. New KS regularisation
3. KS termination
4. Output
5. 3-body regularisation
6. 4-body regularisation
7. New hierarchical system
8. Termination of hierarchical system
9. Chain regularisation
10. Physical collisions



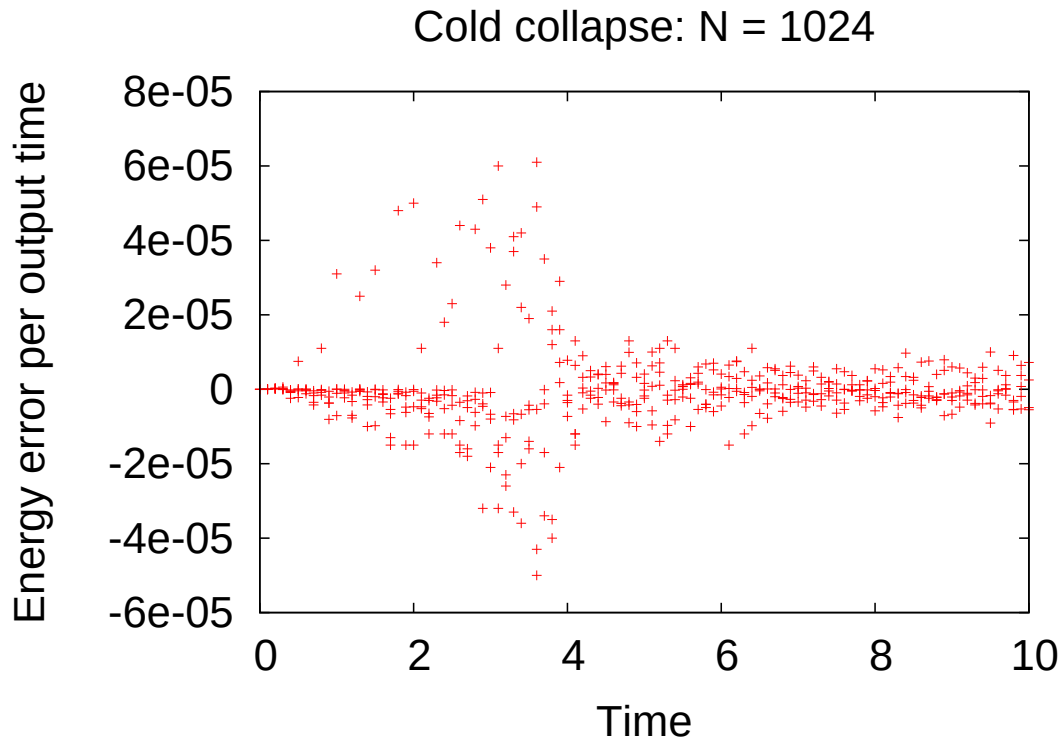
Also: stellar evolution

N-body codes

1. NBODY1-6 (Aarseth), NBODY6++ (Spurzem): FORTRAN, production code
2. starlab (McMillan, Hut, Makino, Portegies Zwart): C++, no KS regularisation, production code
3. ACS (Hut, Makino): Ruby, experimental
4. MUSE (everyone): Python, FORTRAN, C++, C, experimental; includes a Hermite code, NBODY1h, and a Barnes-Hut tree code

Quality control

- Energy check



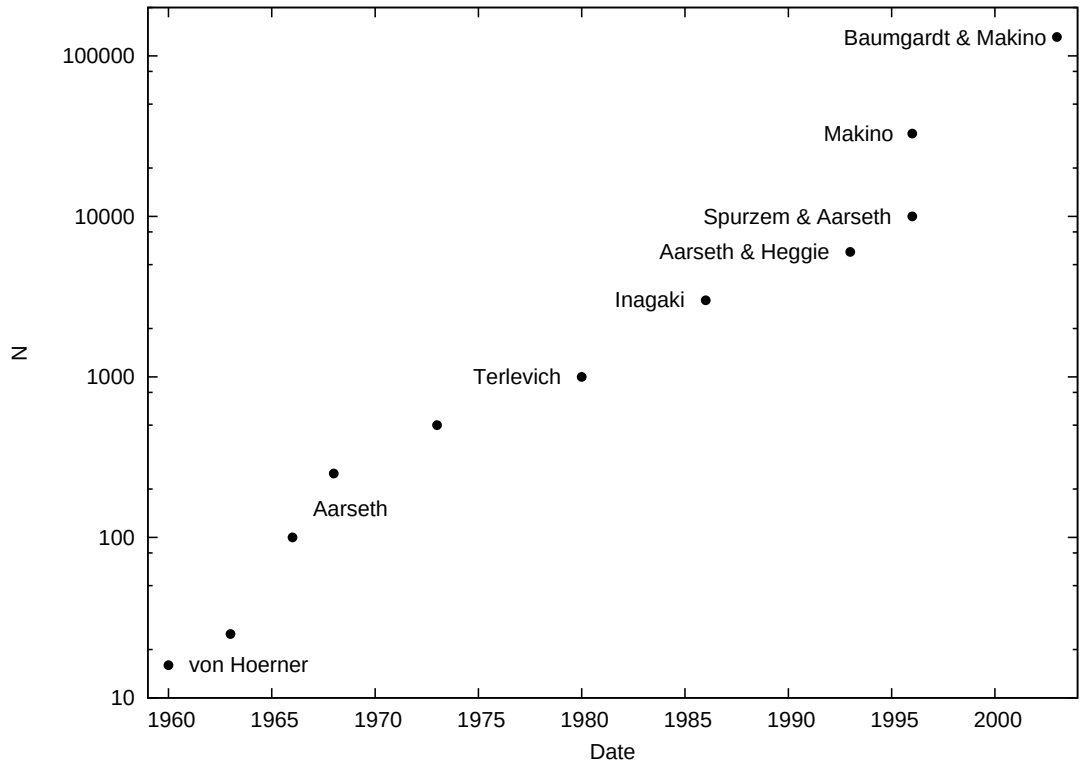
- Are your answers reasonable?

"Complexity"

- Effort dominated by calculation of $\mathbf{a}$: N terms
- N forces to be calculated
- Typical time step $\ll t_{cr}$
- Typical collisional simulation lasts few $t_r \sim N t_{cr}$

Hence effort $\propto N^3$ at least.

The Development of N-body Simulations in History



Something else is needed for large N.

Monte Carlo codes

Assume

- Spherical symmetry
- Dynamic equilibrium

In a given potential, each particle characterised by

- energy E
- angular momentum L

These evolve on a time scale of t_r. Choose time step $= \eta t_{rh}$.

A Monte Carlo Algorithm

1. Initialise positions and velocities, compute E_i, L_i
2. Order the particles by radius, and compute gravitational potential ϕ (easy in spherical symmetry)
3. For each successive pair $i, i+1$ compute the local value of t_r, and let $i, i+1$ have a two-body encounter yielding, on average, the correct effect of two-body relaxation in the time interval Δt.
4. Calculate new E, L for all particles
5. Reassign radii of all particles (according to the time spent at each radius given E, L, ϕ)
6. Repeat from 2

Refinements to the Monte Carlo Algorithm

- Different time steps in different zones
- Binaries and their interactions using
  - cross sections (Giersz & Heggie)
  - on-the-fly few-body integrations (Fregeau)
- Stellar evolution (e.g. McScatter interface)

Recent applications (see talks at Capri)

- open cluster M67
- globular cluster M4

References

- Stellar Dynamics
  1. *Galactic Dynamics* James Binney and Scott Tremaine, Princeton University Press, 1988, £38.95, 755 pp.
  2. *Dynamical Evolution of Globular Clusters* Lyman J. Spitzer Jr, Princeton UP, 1988, out of print, 196 pp.
  3. *The Gravitational Million Body Problem*, Douglas Heggie, Piet Hut; Cambridge UP, 2003, $65.00
- *N*-body codes
  1. *Gravitational N-body simulations*, Sverre J. Aarseth Cambridge: CUP, 2003, £49, 413pp
  2. http://www.ids.ias.edu/∼starlab/ (starlab)
  3. http://artcompsci.org/ (*The Art of Computational Science*)
  4. Hénon, M. 1971, *The Monte Carlo Method*, Ap&SS, 14, 151