

# $N$ -Body Methods and Algorithms

Sverre Aarseth

Institute of Astronomy, Cambridge

Equations of Motion

Hermite Method

Basic Hermite Code

Neighbour Scheme

NBODY2H

Programming Exercises

<http://www.ast.cam.ac.uk/~sverre>

<http://www.sverre.com>

# Newton's Equations

$$\text{Force} \quad \mathbf{F}_i = -G \sum_{j=1; j \neq i}^N m_j \frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|^3}$$

Explicit differentiation

$$\begin{aligned} \mathbf{F}_i^{(1)} = & -G \sum_{j=1; j \neq i}^N m_j \frac{\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_j}{|\mathbf{r}_i - \mathbf{r}_j|^3} \\ & - 3m_j \frac{(\mathbf{r}_i - \mathbf{r}_j) \cdot (\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_j)}{|\mathbf{r}_i - \mathbf{r}_j|^2} \frac{\mathbf{r}_i - \mathbf{r}_j}{|\mathbf{r}_i - \mathbf{r}_j|^3} \end{aligned}$$

New solution at  $t = \Delta t$

$$\Delta \dot{\mathbf{r}}_i = \left( \frac{1}{2} \mathbf{F}_i^{(1)} \Delta t + \mathbf{F}_i \right) \Delta t$$

$$\Delta \mathbf{r}_i = \left( \left( \frac{1}{6} \mathbf{F}_i^{(1)} \Delta t + \frac{1}{2} \mathbf{F}_i \right) \Delta t + \dot{\mathbf{r}}_i \right) \Delta t$$

# Softening

Potential  $\Phi(r) = \frac{Gm}{(r^2 + \epsilon^2)^{1/2}}$

Energy  $E = \frac{1}{2}v^2 - \Phi(r) = \text{const}$

Force  $\mathbf{F}_i = -G \sum_{j=1; j \neq i}^N m_j \frac{\mathbf{r}_i - \mathbf{r}_j}{(|\mathbf{r}_i - \mathbf{r}_j|^2 + \epsilon^2)^{3/2}}$

Force derivative

$$\mathbf{F}_i^{(1)} = -G \sum_{j=1; j \neq i}^N m_j \frac{\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_j}{(|\mathbf{r}_i - \mathbf{r}_j|^2 + \epsilon^2)^{3/2}} - 3m_j \frac{(\mathbf{r}_i - \mathbf{r}_j) \cdot (\dot{\mathbf{r}}_i - \dot{\mathbf{r}}_j)}{|\mathbf{r}_i - \mathbf{r}_j|^2 + \epsilon^2} \frac{\mathbf{r}_i - \mathbf{r}_j}{(|\mathbf{r}_i - \mathbf{r}_j|^2 + \epsilon^2)^{3/2}}$$

# Hermite Integration

Taylor series for  $\mathbf{F}$  and  $\mathbf{F}^{(1)}$

$$\mathbf{F} = \mathbf{F}_0 + \mathbf{F}_0^{(1)} t + \frac{1}{2} \mathbf{F}_0^{(2)} t^2 + \frac{1}{6} \mathbf{F}_0^{(3)} t^3$$

$$\mathbf{F}^{(1)} = \mathbf{F}_0^{(1)} + \mathbf{F}_0^{(2)} t + \frac{1}{2} \mathbf{F}_0^{(3)} t^2$$

Prediction

$$\mathbf{r}_j = \left( \left( \frac{1}{6} \mathbf{F}_0^{(1)} \delta t'_j + \frac{1}{2} \mathbf{F}_0 \right) \delta t'_j + \mathbf{v}_0 \right) \delta t'_j + \mathbf{r}_0$$

$$\mathbf{v}_j = \left( \left( \frac{1}{2} \mathbf{F}_0^{(1)} \delta t'_j + \mathbf{F}_0 \right) \delta t'_j + \mathbf{v}_0 \right); \quad \delta t'_j = t - t_0$$

New forces  $\mathbf{F}, \mathbf{F}^{(1)}$

Higher derivatives

$$\mathbf{F}_0^{(3)} = (2(\mathbf{F}_0 - \mathbf{F}) + (\mathbf{F}_0^{(1)} + \mathbf{F}^{(1)}) t) \frac{6}{t^3}$$

$$\mathbf{F}_0^{(2)} = (-3(\mathbf{F}_0 - \mathbf{F}) - (2\mathbf{F}_0^{(1)} + \mathbf{F}^{(1)}) t) \frac{2}{t^2}$$

Corrector for  $i$

$$\Delta \mathbf{r}_i = \frac{1}{24} \mathbf{F}_0^{(2)} \Delta t^4 + \frac{1}{120} \mathbf{F}_0^{(3)} \Delta t^5$$

$$\Delta \mathbf{v}_i = \frac{1}{6} \mathbf{F}_0^{(2)} \Delta t^3 + \frac{1}{24} \mathbf{F}_0^{(3)} \Delta t^4$$

# Basic Code Structure

Input	Read input parameters
Initial conditions	Generate $m, \mathbf{r}, \dot{\mathbf{r}}$
Initialization	$\mathbf{F}, \mathbf{F}^{(1)}$ & $\Delta t$
Scheduling	Form block-step distribution
Prediction	Neighbours or all $N$
Neighbour integration	Sequential $\mathbf{F}_n$ & $\mathbf{F}_n^{(1)}$
Regular force calc	Sequential total force
New block-step	Determine next group

# Time-Steps

Basic time-step  $\Delta t = \frac{\alpha|\mathbf{r}|}{|\mathbf{v}|}, \quad \Delta t = \frac{\beta|\mathbf{F}|}{|\mathbf{F}^{(1)}|}$

Taylor series  $\mathbf{F} = \mathbf{F}_0 + \mathbf{F}_0^{(1)} \Delta t + \frac{1}{2}\mathbf{F}_0^{(2)} \Delta t^2 + \dots$

Natural time-step  $\Delta t = \left( \frac{\eta|\mathbf{F}|}{|\mathbf{F}^{(2)}|} \right)^{1/2}, \quad \eta = 0.02$

General expression  $\Delta t = \left( \frac{\eta(|\mathbf{F}||\mathbf{F}^{(2)}| + |\mathbf{F}^{(1)}|^2)}{|\mathbf{F}^{(1)}||\mathbf{F}^{(3)}| + |\mathbf{F}^{(2)}|^2} \right)^{1/2}$

Relative criterion  $\Delta t$  independent of mass

Block-steps  $\Delta t_n = \frac{\Delta t_1}{2^{n-1}}, \quad \Delta t_1 = 1$

Hierarchical levels  $\mathcal{N}_k$  particles with steps  $\Delta t_k$

Scheduling  $i = \min (t_j + \Delta t_j)$

# Units

## (a) Scaling relations

Given length scale  $R_V$  in pc and total mass  $NM_S$  in  $M_\odot$

Velocity scaling

$$\tilde{V}^* = 1 \times 10^{-5} (GM_\odot/L^*)^{1/2} \text{ km/s, with } L^* = 3 \times 10^{18} \text{ cm}$$

$$\text{Velocity unit} \quad V^* = 6.557 \times 10^{-2} (NM_S/R_V)^{1/2} \text{ km/s}$$

$$\text{Fiducial time} \quad \tilde{T}^* = (L^{*3}/GM_\odot)^{1/2} = 14.94 \text{ Myr}$$

$$\text{Time unit} \quad T^* = 14.94 (R_V^3/NM_S)^{1/2} \text{ Myr}$$

## (b) Conversion from N-body to physical units

$$\tilde{r} = R_V r \text{ pc, } \tilde{v} = V^* v \text{ km/s, } \tilde{t} = T^* t \text{ Myr,}$$
$$\tilde{m} = NM_S m M_\odot$$

$$\text{Crossing time} \quad T_{\text{cr}} = 2\sqrt{2} T^* \text{ Myr}$$

# Neighbour Scheme

Total force  $\mathbf{F}(t) = \sum_{j=1}^n \mathbf{F}_j + \mathbf{F}_d(t)$

Prediction

$$\mathbf{F}(t) = \mathbf{F}_n + \dot{\mathbf{F}}_d(t - t_0) + \mathbf{F}_d(t_0)$$

$$\dot{\mathbf{F}} = \dot{\mathbf{F}}_n + \dot{\mathbf{F}}_d$$

Time-scales

$$\Delta t_n \ll \Delta t_d, \quad n \ll N$$

Neighbour sphere  $R_s^{\text{new}} = R_s^{\text{old}} \left( \frac{n_p}{n} \right)^{1/3}$

Neighbour selection  $|\mathbf{r}_i - \mathbf{r}_j| < R_s$

Derivative corrections  $\mathbf{F}_{ij}^{(2)}, \mathbf{F}_{ij}^{(3)}$



# Modification of COMMON

(a) Constant size

Existing dummies      ..., *XDUM*(10), *NDUM*(10)

New variables              *XNEW*(2), *NEW*

..., *XNEW*(2), *XDUM*(8), *NEW*, *NDUM*(9)

(b) Enlargement

Increase COMMON      *COMMON/EXTRA/ A*(5), *B*, *NEW*(6)

Add to MYDUMP        *REAL \* 4 XNEW*

New COMMON            *COMMON/EXTRA/ XNEW*(18)

Add READ/WRITE      ..., *XNEW*

# Programming

NBODY6	330 routines and 56,000 lines
Languages	FORTRAN, C++ & CUDA
Basic variables	70 size $N$ arrays & size $N$ lists
Parameters	$NMAX$ before compilation
Common blocks	common6.h, arrays & scalars
Subroutines	FORTRAN, C++ & GPU
Coding style	6 letter variables, verbal comments
Options	More than 40, multiple values
Downloads	<a href="http://www.ast.cam.ac.uk/research/nbody">www.ast.cam.ac.uk/research/nbody</a>

May 25, 12 12:29

## Makefile

Page 1/1

```

SOURCE = \
nbody6.f adjust.f assess.f bindat.f binev.f binout.f binpop.f \
block.f bodies.f brake.f brake2.f brake3.f bsetid.f chaos0.f chaos.f \
check.f checkl.f chrect.f clint.f cloud.f cloud0.f cmbody.f \
cmcorr.f cmfirr.f cmfreg.f coal.f comenv.f core.f corerd.f \
cputim.f data.f decide.f define.f deform.f degen.f delay.f \
dgc core.f dtchck.f eccmod.f ecirc.f edot.f efac2.f efac3.f \
expel.f energy.f escape.f events.f evolve.f expand.f fclose.f \
fcloud.f fcorr.f fdisk.f fhalo.f ficorr.f findj.f findm.f \
flyby.f fnuc.f fpcorr.f fpert.f fpoly1.f fpoly2.f freeze.f \
gcinit.f gcint.f giant.f giant3.f gntage.f ggrad.f hcorr.f \
hiarch.f hicirc.f hidat.f higrow.f himax.f himax2.f himod.f \
hipop.f hirect.f histab.f hivel.f hmdot.f hmdot2.f hotsys.f \
hrdiag.f hrplot.f hut.f hut2.f iblock.f imf.f imfbd.f imf2.f \
impact.f induce.f inext.f input.f insert.f instar.f intgrt.f \
intide.f jacobi.f kepler.f kick.f kick2.f ksapo.f kscorr.f \
ksin2.f ksinit.f ksint.f kslist.f ksmod.f ksperi.f kspert.f \
kspoly.f kspred.f ksrect.f ksreg.f ksres.f ksres2.f ksterm.f \
kstide.f lagr.f lagr2.f levels.f magbrk.f matrix.f mdot.f merge.f \
merge2.f mix.f mloss.f mlwind.f modify.f mrenv.f mtrace.f mydump.f \
nbhist.f nbint.f nblist.f nbpot.f nbrem.f nbrest.f nbsort.f nbtide.f \
newtev.f nstab.f ntint.f offset.f orbit.f output.f peri.f permit.f \
pfac.f poti.f proto_star.f qtides.f ran2.f reflct.f regint.f \
remove.f rename.f reset.f reset2.f resolv.f rkint.f rl.f roche.f \
rpmax.f rpmax2.f rpmin.f scale.f search.f setup.f setup2.f shrink.f \
sort1.f spiral.f stability.f star.f start.f stepk.f steps.f stumpf.f \
subint.f swap.f sweep.f synch.f tail0.f tcirc.f tides.f tides2.f \
tides3.f touch.f tpert.f trdot.f trdot2.f trflow.f tstab.f tstep.f \
units.f unpert.f update.f verify.f xtrnl0.f xtrnld.f xtrnlf.f xtrnlp.f \
xtrnlt.f xtrnlv.f xvpred.f zare.f zcnsts.f zero.f zfuncs.f \
triple.f derqp3.f difsy3.f erel3.f extend.f qpmod3.f stabl3.f \
stablz.f start3.f subsys.f tperi.f trans3.f \
quad.f derqp4.f difsy4.f endreg.f erel4.f ichain.f newreg.f newsys.f \
qpmod4.f rchain.f rsort.f stabl4.f start4.f status.f trans4.f \
cfuncs.f chain.f chstab.f const.f cstab2.f cstab3.f cstab4.f cstab5.f \
derqp.f difsy1.f erel.f hpsort.f inclin.f invert.f ksphys.f physks.f \
qforce.f qpmod.f r2sort.f recoil.f redraw.f select.f slow.f stablc.f \
swcond.f switch.f transk.f transq.f transx.f vector.f xtf.f xtrnlu.f \
ycopy.f ysave.f \
absorb.f chaos2.f chdata.f chfind.f chfirr.f chinit.f chlist.f chmod.f \
chpert.f chpot.f chterm.f expel2.f fchain.f ghost.f giant2.f kcpert.f \
reduce.f reinit.f renew.f setsys.f tchain.f xcpred.f xtpert.f premsf.f \
circ.f spinup.f

```

# Code Tests

Chaotic system      Exponential error growth

Small time-steps       $\Delta t \propto R^{3/2}$

Two-body problem      Analytical solution

Evolution rate      Core collapse time

Mass segregation      Two mass groups

Systematic errors      Minimize energy drift

Performance      CPU time as  $f(N)$