falcON

a Cartesian FMM for the low-accuracy regime

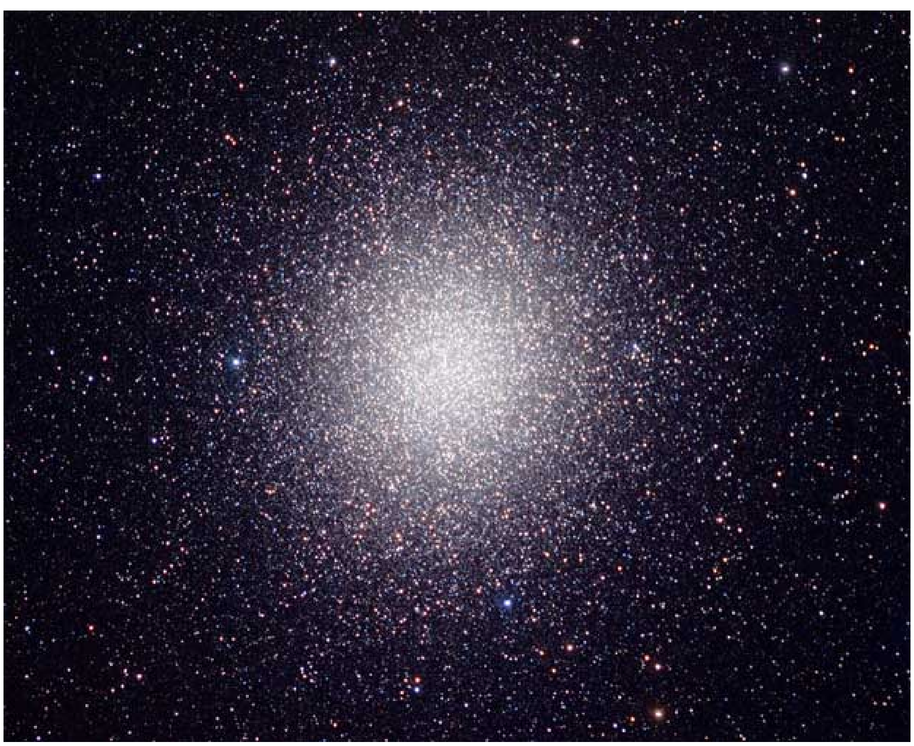
College Park, 28^{th} April 2004

Journal of Computational Physics, 179, 27-42 (2002)

Walter Dehnen (Leicester)
walter.dehnen@astro.le.ac.uk

N-body simulations in astronomy

HCG87: a group of galaxies



ωCen: a globular cluster

properties of stellar systems

- ▷ simple physics: Newtonian gravity
- ▷ very inhomogeneous
  ⇒ large dynamic range
- ▷ dynamically young ($t_{\mathrm{dyn}} \simeq$ Myr–Gyr)
- ▷ well approximated as ensembles of point masses
  ⇒ well described as Hamiltonian systems
  (⇒ need symplectic time integration)

$$\mathrm{H} = \sum_{i=1}^{N} \frac{m_i}{2} \left[\boldsymbol{v}_i^2 - \sum_{j \neq i} \frac{G\, m_j}{|\boldsymbol{x}_i - \boldsymbol{x}_j|} \right], \qquad \boldsymbol{v}_i = \dot{\boldsymbol{x}}_i = \frac{\boldsymbol{p}_i}{m_i}$$

with $N \simeq 10^{5-20}$

- ▷ equation of motion in continuum (mean-field) limit:

$$0 = \frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \boldsymbol{x}} \cdot \boldsymbol{v} - \frac{\partial f}{\partial \boldsymbol{v}} \cdot \frac{\partial \Phi}{\partial \boldsymbol{x}}$$

collisionless Boltzmann equation (**CBE**)

- ▷ $f(\boldsymbol{x}, \boldsymbol{v}, t)$: **distribution function** (density in phase space)
- ▷ $\Phi(\boldsymbol{x})$: mean-field **gravitational potential**

both are related via the **Poisson equation**:

$$\nabla^2 \Phi(\boldsymbol{x}) = 4\pi G \int \mathrm{d}^3\boldsymbol{v}\, f(\boldsymbol{x}, \boldsymbol{v}, t)$$

two-body relaxation

How good is the continuum description?

▷ stellar encounters deflect trajectories
  ⇒ stellar orbits get randomized
  ⇒ Maxwellian velocity distribution

▷ **two-body relaxation time**:

$$t_{\mathrm{relax}} \simeq 0.1 \frac{N}{\log N} t_{\mathrm{dyn}}$$

1 collision-dominated stellar dynamics

▷ $t_{\mathrm{relax}} \lesssim$ age of system

⇒ continuum limit not applicable
⇒ must simulate Hamiltonian directly:
  ▷ force computation is $\mathcal{O}(N^2)$
    ⇒ computational effort limits $N \lesssim 10^5$
  ▷ close encounters are important
    ⇒ time integration becomes tedious

2 collisionless stellar dynamics

▷ $t_{\mathrm{relax}} \gg$ age of system

⇒ continuum limit applicable
⇒ solve **CBE** & **Poisson equation**

‘collisionless’ N-body simulations

How to solve the CBE?

$$0 = \frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t} + \frac{\partial f}{\partial \boldsymbol{x}} \cdot \boldsymbol{v} - \frac{\partial f}{\partial \boldsymbol{v}} \cdot \frac{\partial \Phi}{\partial \boldsymbol{x}}$$

▷ f is 6D & very inhomogeneous

⇒ (Eulerian) grid methods are useless

⇒ Lagrangian method (‘method of characteristics’):

▷ sample N trajectories $\{\mu_i, \boldsymbol{x}_i, \boldsymbol{v}_i\}$ from $f(\boldsymbol{x}, \boldsymbol{v}, t = 0)$

▷ solve equations of motion $\ddot{\boldsymbol{x}}_i = -\boldsymbol{\nabla}\Phi(\boldsymbol{x}_i, t)$

▷ CBE: $\mu_i =$ const along trajectories

⇒ $f(\boldsymbol{x}, \boldsymbol{v}, t)$ is represented by $\{\mu_i, \boldsymbol{x}_i(t), \boldsymbol{v}_i(t)\}$

⇒ f is *unknown*

⇒ **moments** of f can be *estimated*

⇒ N $\ll N$ is *numerical parameter*

⇒ artificial two-body relaxation

How to solve the Poisson equation?

$$\nabla^2 \Phi(\boldsymbol{x}) = 4\pi G \int \mathrm{d}^3\boldsymbol{v}\, f(\boldsymbol{x}, \boldsymbol{v}, t)$$

1 grid techniques (FFT, multigrid):

▷ fast: $\mathcal{O}(n_{\text{grid}} \log n_{\text{grid}})$

▷ periodic (⇒ cosmology)

▷ problem: inhomogeneity (but: adaptive multigrid)

2 basic functions (using Y_{lm}):

▷ fast: $\mathcal{O}(\mathrm{N}\, n_{\text{basis}})$

▷ problems: central singularity, spherical symmetry

3 Greens-function approach:

$$\Phi(\boldsymbol{x}, t) = -G \int \mathrm{d}^3\boldsymbol{x}'\, \mathrm{d}^3\boldsymbol{v}\, \frac{f(\boldsymbol{x}', \boldsymbol{v}, t)}{|\boldsymbol{x} - \boldsymbol{x}'|}$$

▷ general & adaptive

▷ problem: f is unkown

⇒ *estimate* (ϵ: **softening length**)

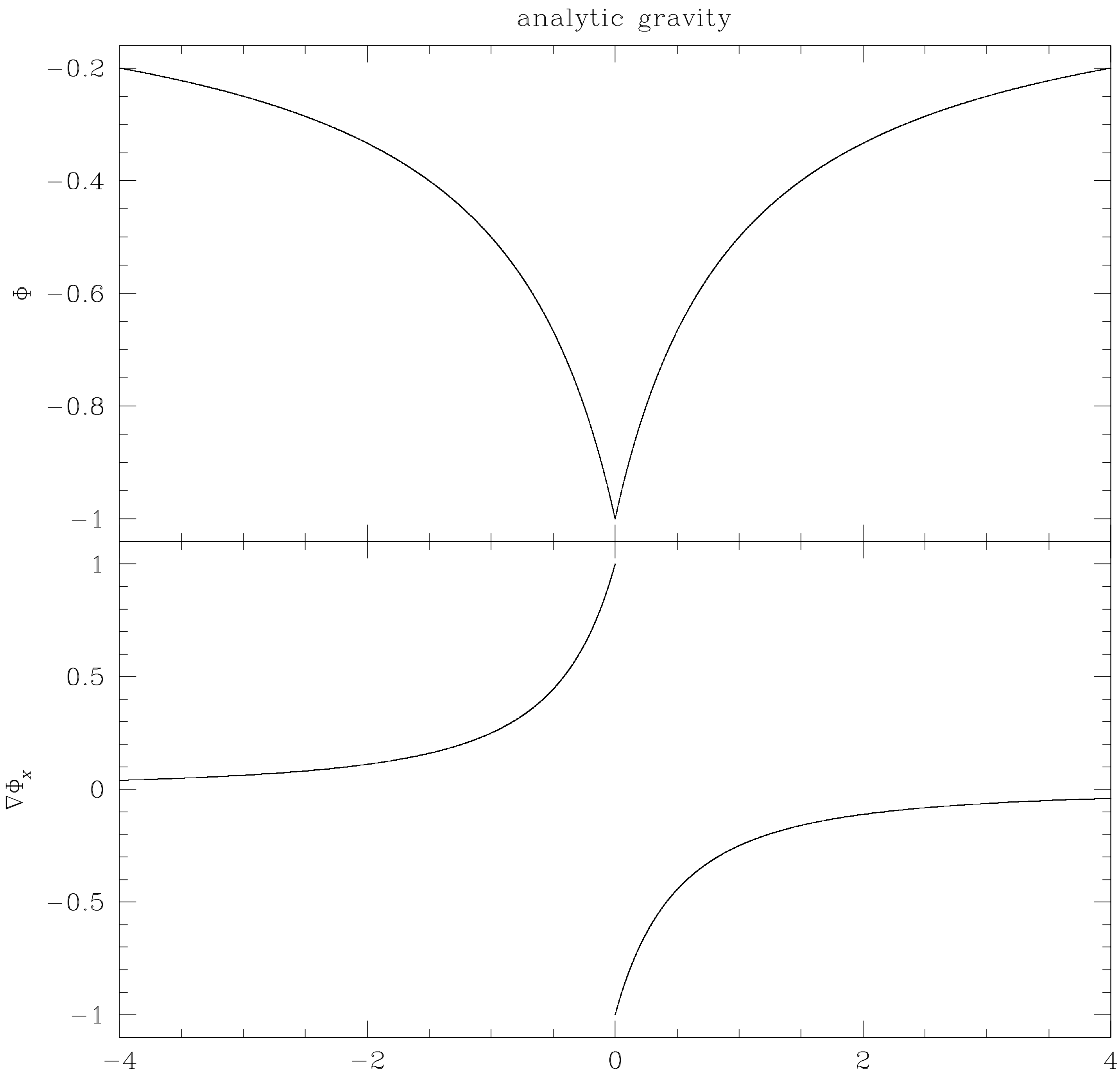
$$\Phi(\boldsymbol{x}_i, t) \approx -\sum_{i \neq j} \frac{G\, \mu_j}{\sqrt{[\boldsymbol{x}_i - \boldsymbol{x}_j(t)]^2 + \epsilon^2}}$$

force softening to

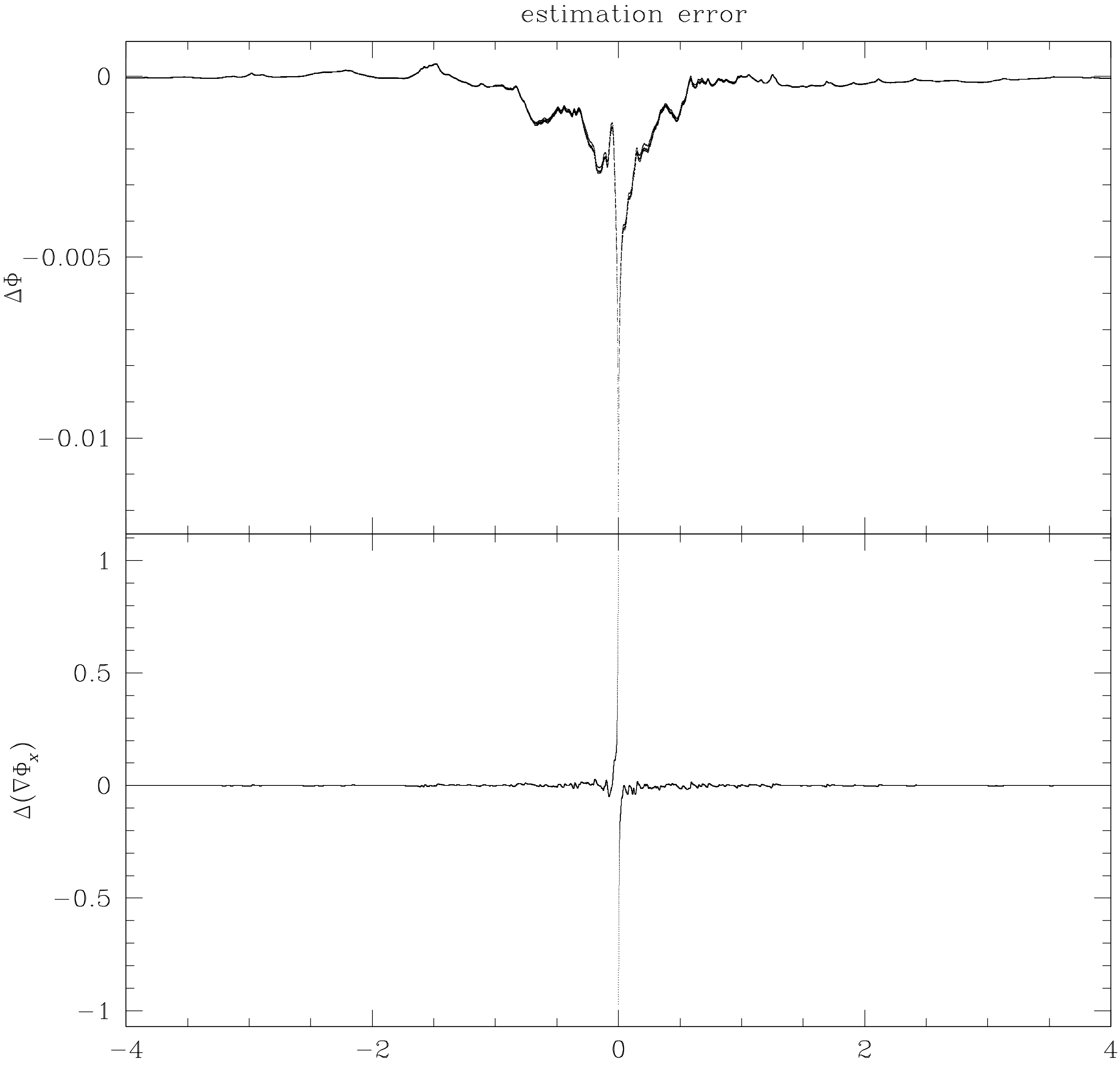
▷ optimize force estimate (since f is unknown)

▷ suppress (unphysically) close encounters

⇒ force-**estimation error** (unavoidable)



true gravity of Hernquist model



estimation error with $N = 10^6$

computing the forces

▷ Greens-function approach → Hamiltonian:

$$\mathsf{H} = \sum_{i=1}^{\mathsf{N}} \frac{\mu_i}{2} \left[\boldsymbol{v}_i^2 - \sum_{j \neq i} \frac{G\,\mu_j}{\sqrt{|\boldsymbol{x}_i - \boldsymbol{x}_j|^2 + \epsilon^2}} \right]$$

▷ how to evaluate Φ & $\boldsymbol{\nabla}\Phi$?

▷ can tolerate **approximation error** ≪ **estimation error**

⇒ use **approximative** methods

1 direct summation (not approximative):

- ▷ slow: $\mathcal{O}(\mathsf{N}^2)$ (but: GRAPE)
- ▷ (unnecessarily) accurate
- ▷ used in *collisional* N-*body* codes

2 Barnes & Hut (1986) **tree code**:

- ▷ use hierarchical tree (usually: oct-tree) ⇒ fully adaptive
- ▷ fast(er): $\mathcal{O}(\mathsf{N}\log\mathsf{N})$
- ▷ most common method in astrophysics
- ▷ violates Newton's 3rd law

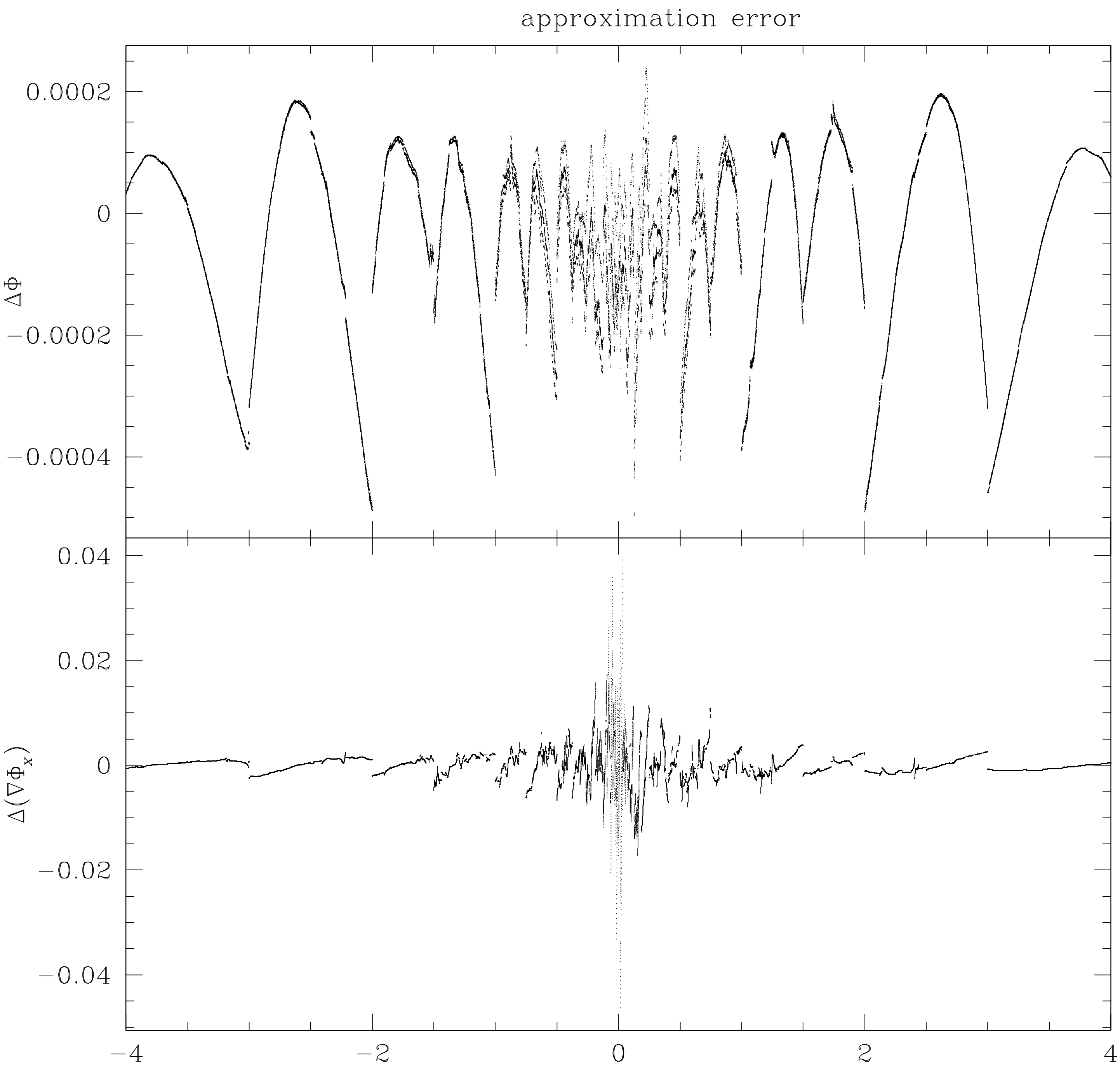
  ⇒ total momentun not conserved

3 traditional **fast multipole method** (**FMM**):

- ▷ use hierarchy of cartesian grids ⇒ not fully adaptive
- ▷ compute gravity via spherical multipoles & complex Y_{lm}

  ⇒ numerics complicated & cumbersome
- ▷ formally $\mathcal{O}(\mathsf{N})$, **but**

  slower than **tree code** (for astrophyiscal applications, see Capuzzo-Colcetta & Miochi, 1998, JCP, **143**, 29)



approximation error with $N = 10^6$

details of the tree code

1 **preparation phase**

1.1 build a hierarchical tree of cubic cells

- ▷ cost: $\mathcal{O}(N \log N)$

1.2 pre-compute multipole moments etc

2 force computation: **'tree-walk'**

- ▷ for each body: compute force due to root cell
- ▷ to compute force from cell:

  if body is **well-separated** from cell:

  compute force from multipole moments

  otherwise

  sum forces from daughter cells (recursive)
- ▷ cost: $\mathcal{O}(\log N)$ per body $\Rightarrow$ $\mathcal{O}(N \log N)$

▷ the tree code is wasteful:

forces of neighbours are similar yet independently computed

details of the FMM

here I describe traditional Greengard & Rokhlin (1987) FMM

1 **preparation phase**

1.1 build a hierarchy of cartesian grids

1.2 pre-compute multipole moments etc (**upward pass**)

2 force computation

2.1 **interactions**

on each grid level:

- ▷ perform ‘intermediate-field’ interactions:
  compute & accumulate multipoles of gravity field

2.2 **downward pass**

- ▷ pass field-multipoles down the hierarchy
- ▷ compute forces on finest grid

- ▷ theoretical $\mathcal{O}(N)$ not demonstrated in practice
- ▷ not competetive with tree code in low-accuracy regime

details of falcON

▷ hybrid of tree code & FMM

▷ takes the better of each method

1 **preparation phase** (as for tree code)

1.1 build a hierarchical tree of cubic cells

▷ cost: $\mathcal{O}(N \log N)$

1.2 pre-compute multipole moments etc

2 force computation

2.1 **interaction phase**

▷ 'catch' all body-body interactions in well-separated node-node interactions:

- if node-node interaction is executable
  execute it: accumulate field tensors
- otherwise
  split it & continue with child interations (recursive)

▷ cost: (better than) $\mathcal{O}(N)$, dominates

2.2 **evaluation phase**

▷ pass field tensors down the tree

▷ compute forces at body positions

▷ cost: $\mathcal{O}(N)$

▷ ~ 10 times faster than tree code or FMM (at low accuracy)

numerics of falcON

Wanted:

$$\Phi(\boldsymbol{x}_i) = -\sum_{j\neq i} \mu_j\, g(\boldsymbol{x}_i - \boldsymbol{y}_j),$$

Taylor expand g about $\boldsymbol{R} = \boldsymbol{x}_0 - \boldsymbol{y}_0$

$$g(\boldsymbol{x}-\boldsymbol{y}) = \sum_{n=0}^{p} \frac{1}{n!}(\boldsymbol{x}-\boldsymbol{y}-\boldsymbol{R})^{(n)} \odot \boldsymbol{\nabla}^{(n)} g(\boldsymbol{R}) \;+\; \mathcal{R}_p(g),$$

Insert & sum over source cell B

$$\Phi_{\mathrm{B}\to\mathrm{A}}(\boldsymbol{x}) = -\sum_{m=0}^{p} \frac{1}{m!}(\boldsymbol{x}-\boldsymbol{x}_0)^{(m)} \odot \mathbf{C}^{m,p} \;+\; \mathcal{R}_p(\Phi_{\mathrm{B}\to\mathrm{A}})$$

$$\mathbf{C}^{m,p} = \sum_{n=0}^{p-m} \frac{(-1)^n}{n!} \boldsymbol{\nabla}^{(n+m)} g(\boldsymbol{R}) \odot \mathbf{M}^n_{\mathrm{B}},$$

$$\mathbf{M}^n_{\mathrm{B}} = \sum_{\boldsymbol{y}_i\in\mathrm{B}} \mu_i\,(\boldsymbol{y}_i - \boldsymbol{y}_0)^{(n)}.$$

(Warren & Salmon 1995: Comp. Phys. Comm, 87, 266)

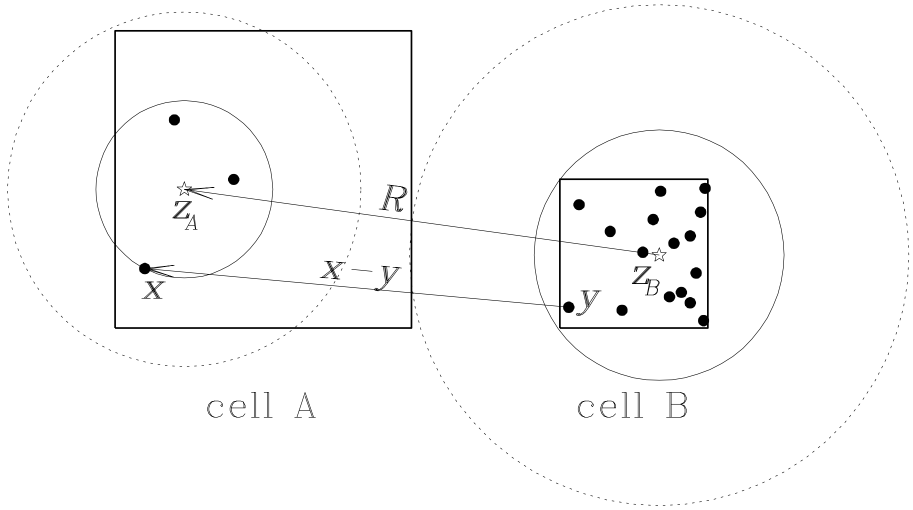
$\sum_m$: evaluation of gravity, represented by the **field tensors** $\mathbf{C}^{m,p}$, at position $\boldsymbol{x}$

$\sum_n$: interaction between source cell B, represented by the **multipoles** $\mathbf{M}^n_{\mathrm{B}}$, and the sink cell A.

Difference to tree code:

- ▷ **expansion** in $\boldsymbol{x}$ (tree code: $\boldsymbol{x} \equiv \boldsymbol{x}_0$)
- ▷ **mutuality** of **inter**actions

gravity between well-seperated nodes



two well-separated cells

If $|\boldsymbol{R}| > r_{\mathrm{A,crit}} + r_{\mathrm{B,crit}}$ with $r_{\mathrm{crit}} = r_{\mathrm{max}}/\theta$,

$\Rightarrow$ $|\boldsymbol{x}-\boldsymbol{y}-\boldsymbol{R}|<\theta|\boldsymbol{R}|\ \forall\ \boldsymbol{x}\in\mathrm{A},\ \boldsymbol{y}\in\mathrm{B}$ & Taylor series converges

force error of individual interaction:

$$|\nabla\mathcal{R}_p(\Phi_{\mathrm{B}\to\mathrm{A}})| \le \frac{(p+1)\theta^p}{(1-\theta)^2}\,\frac{\mathrm{M_B}}{R^2}$$
$$\propto \frac{\theta^{p+2}}{(1-\theta)^2}\, r_{\mathrm{B,max}}^{d-2} \propto \frac{\theta^{p+2}}{(1-\theta)^2}\, \mathrm{M_B}^{(d-2)/d}$$

▷ standard tree-code & FMM practice: $\theta = \mathrm{const}$
- $\Rightarrow$ **relative** error controlled
- $\Rightarrow$ **absolute** error increases with $\mathrm{M_B}$
- $\Rightarrow$ **total error** dominated by few interactions with large cells

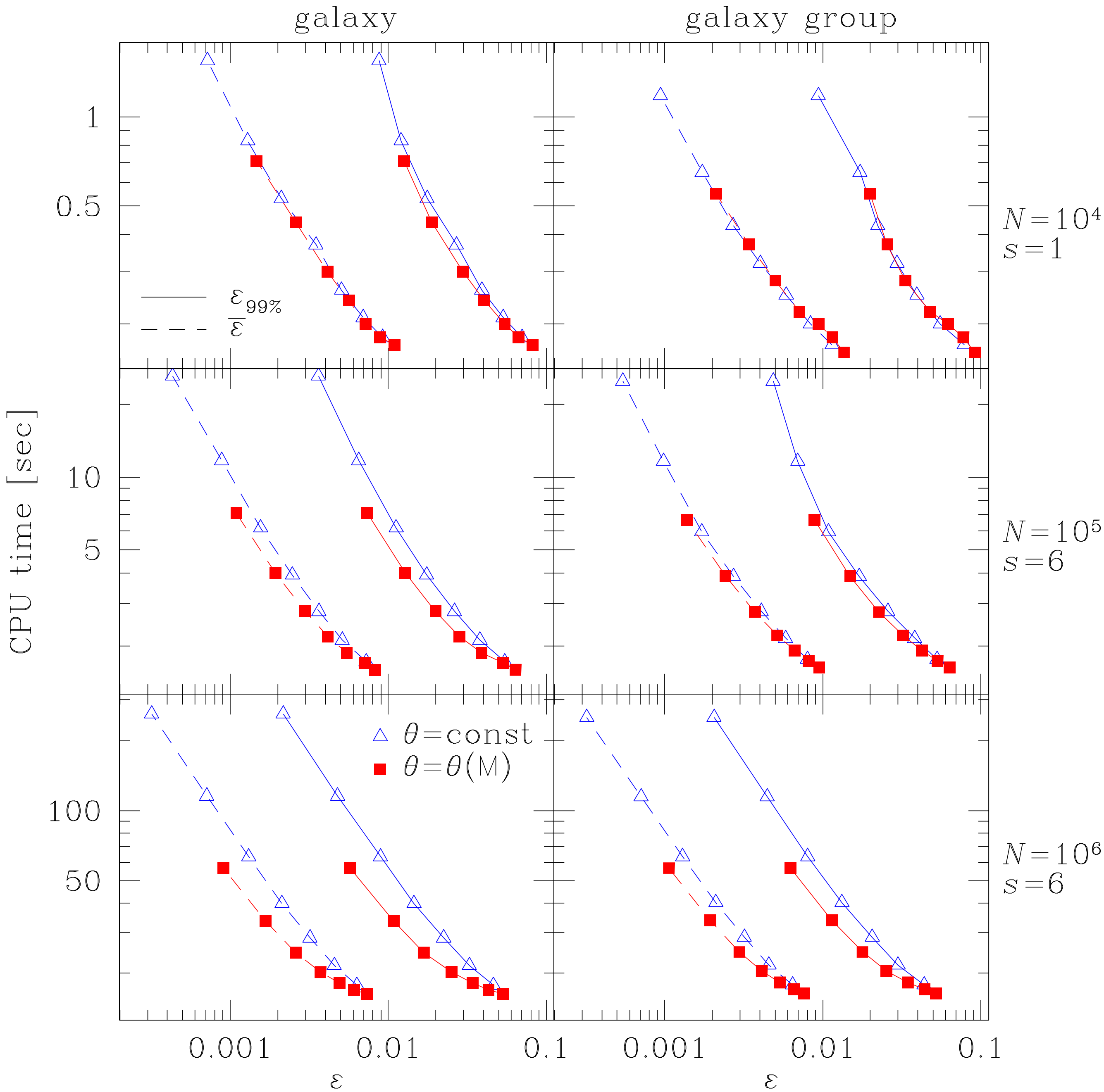
$\Rightarrow$ better:

▷ balance **absolute** individual errors by $\theta = \theta(\mathrm{M})$ with

$$\frac{\theta^{p+2}}{(1-\theta)^2} = \frac{\theta_{\mathrm{min}}^{p+2}}{(1-\theta_{\mathrm{min}})^2}\left(\frac{\mathrm{M}}{\mathrm{M_{tot}}}\right)^{(2-d)/d}$$

$\Rightarrow$ reduce total error

accuracy vs. CPU time

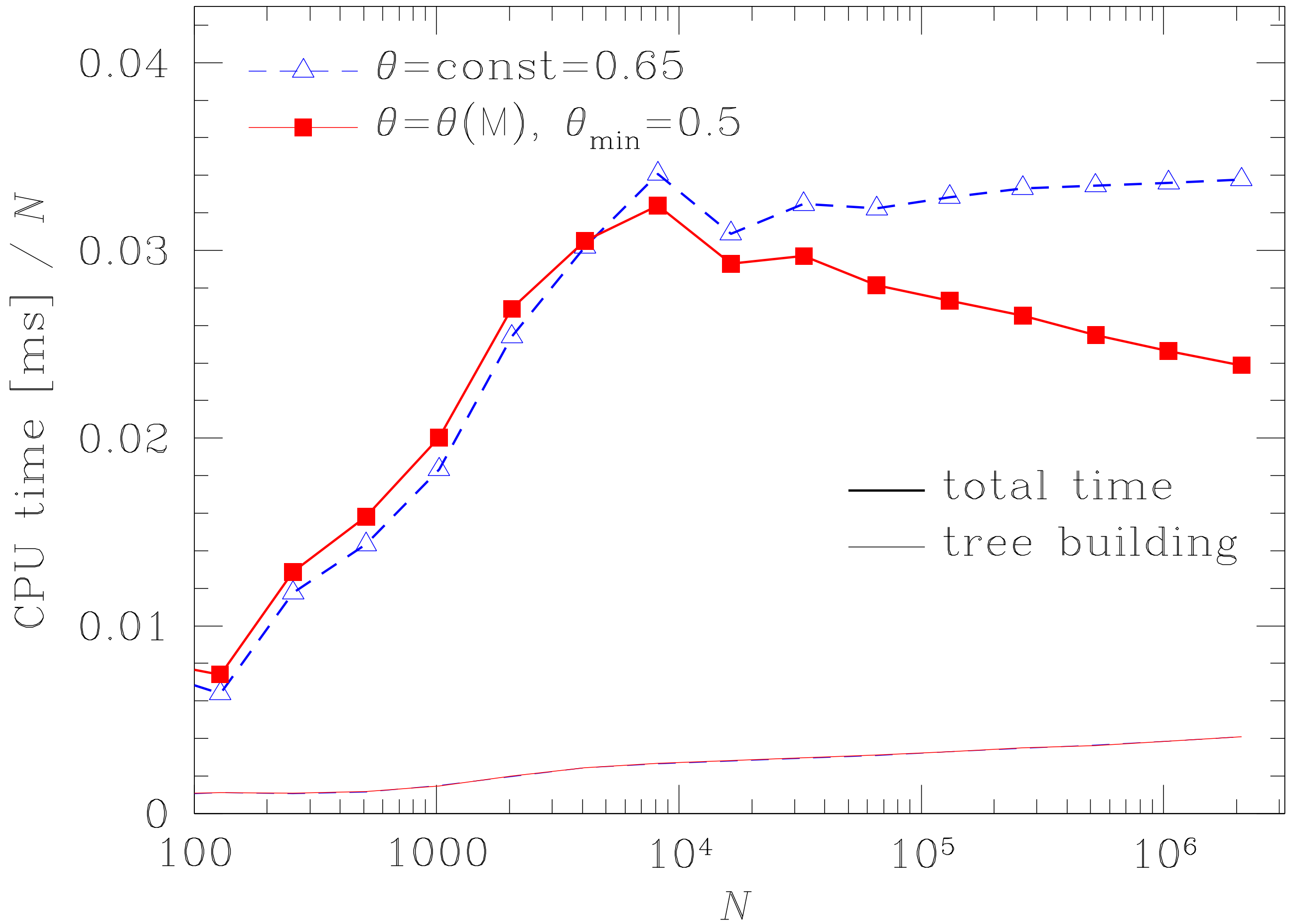


mean (*dashed*) and 99 percentile (*solid*) relative force error

$$\varepsilon \equiv |a_{\mathrm{approx}} - a_{\mathrm{exact}}|/a_{\mathrm{exact}},$$

versus the CPU time (Pentium III/933Mhz in **2001**) for a galaxy (*left*) and a group f galaxies (*right*), sampled with (total) $N = 10^4$ (*top*), $N = 10^5$ (*middle*), or $N = 10^6$ (*bottom*). We used either $\theta = \mathrm{const}$ (*open triangles*) or $\theta = \theta(M)$ (*solid squares*). The symbols along each curve correspond, from left to right, to values for θ or $\theta_{\min}$ of 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, and 0.8.

performance



CPU time per body (Pentium III/500Mhz in **2000**) versus N for a galaxy group.

what complexity?

▷ 8-folding N $\Rightarrow N_I \to 8N_I + N_+$ and thus:

$$\frac{\mathrm{d}N_I}{\mathrm{d}N} \simeq \frac{N_I}{\mathrm{N}}\frac{\Delta \ln N_I}{\Delta \ln N} \approx \frac{N_I}{\mathrm{N}} + \frac{N_+}{N 8 \ln 8},$$

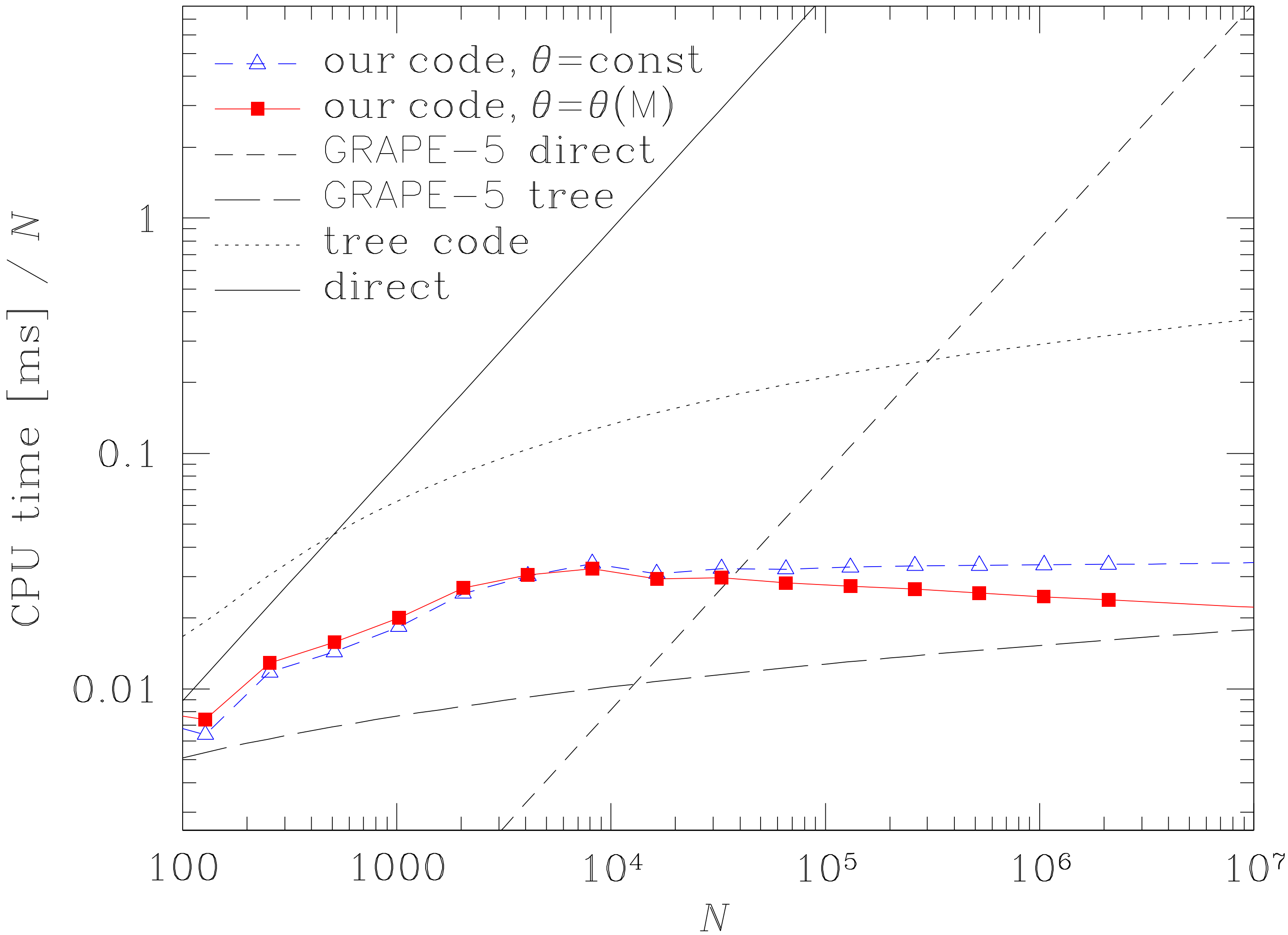
with solution

$$N_I = c_0 N + \frac{N}{8 \ln 8}\int \frac{N_+}{N^2}\,\mathrm{d}N$$

▷ B&H tree code: $N_+ \propto \mathrm{N}$
$\Rightarrow N_I \propto \mathrm{N} \log \mathrm{N}$

▷ Here: $N_+(\mathrm{N})$ grows sub-linear at large N
$\Rightarrow N_I \propto \mathrm{N}$

comparison with other methods used in astrophysics



CPU time per body (**2001**) versus N for various techniques. Note that there are differences in the hard- & software, stellar system, and accuracy requirements.

by **2003/2004**: falcON is ∼ 3 times faster, but GRAPE-5 tree not.

comparison with FMM

▷ comparing under same conditions (bodies uniform in a cube)

$$E = \left[\sum_i \left(\Phi_{i,\mathrm{direct}} - \Phi_{i,\mathrm{approx}} \right)^2 \Big/ \sum_i \Phi^2_{i,\mathrm{direct}} \right]^{1/2}$$

▷ low-accuracy regime: ~ 10 times faster:

timing results (in seconds):

N	T^a_{FMM}	T^a_{direct}	E^a	T^b_{falcON}	T^c_{direct}	E^b
20000	13.3	233	7.9×10^{-4}	0.97	136	3.7×10^{-4}
50000	27.7	1483	5.2×10^{-4}	2.64	924	3.3×10^{-4}
200000	158	24330	8.4×10^{-4}	10.77	14694	3.4×10^{-4}
500000	268	138380	7.0×10^{-4}	29.42	91134	3.7×10^{-4}
1000000	655	563900	7.1×10^{-4}	58.34	366218	3.5×10^{-4}

[a] FMM; data from Table I of Cheng et al. (1999: JCP, 155, 468)

[b] falcON on a computer identical to that used by Cheng et al.

[c] our own implementation of direct summation on the same computer

▷ high-accuracy regime:
falcON cannot compete with FMM

⇒ accuracy & performance depend on both p **&** θ

- ▷ FMM: fixed 'θ', vary p
- ▷ falcON: fixed $p = 3$, vary θ

⇒ high accuracy requires higher order p

summary

- ▷ falcON = hybrid of tree code & FMM
- ▷ new features:
  - ● explicitly exploits mutuality of gravity
    - ⇒ reduces computational effort
    - ⇒ requires novel tree-walking algorithm
    - ⇒ conservation of Newton's 3rd law
  - ● mass-dependent θ
    - ⇒ error balancing
    - ⇒ reduces cost to **better** than $\mathcal{O}(\mathrm{N})$
- ▷ ~ 10 times faster than tree code or FMM
- ▷ publicly available

more dogmas

▷ balance errors

⇒ reduce effort at given accuracy

▷ keep algorithm as simple as possible &

as complicated as necessary

⇒ high-order may be unnecessary

▷ write efficient code

⇒ avoid cache misses

⇒ data structure design

⇒ write generic code

⇒ do not rely too much on compliler optimization

⇒ template metaprogramming