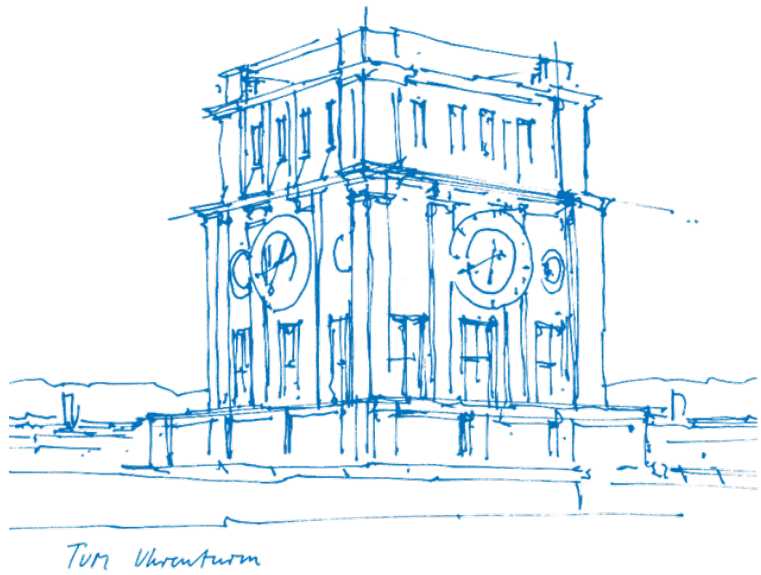
HPC – Algorithms and Applications

Dwarf # 4 – N-Body Methods

Michael Bader
TUM – SCCS

Winter 2016/2017



Computational Effort for N-Body Problems

- to solve: system of ODE

$$\frac{d^2}{dt^2}\vec{r}_i = \frac{\vec{F}_i}{m_i} = \frac{1}{m_i}\sum_{j\neq i}\vec{F}_{ij}$$

- requires forces between all pairs of molecules:

$$\text{“compute } \vec{F}_i = \sum_{j\neq i}\vec{F}_{ij} \text{ for all } i\text{”}$$

- computational effort to compute all forces $\vec{F}_i$ thus $\mathcal{O}(N^2)$
- unfeasible for systems with 10^6 to 10^9 molecules

$\Rightarrow$ **How can the computational effort be reduced?**

Long-Range Interactions

Long-Range Potentials

- examples: Coulomb and gravity potential
- particles in large distance can contribute forces
  (if there are enough of them)
- using cut-off potential leads to large errors
  (all long-distance forces ignored)
- formally: potential decaying slower than r^{-d}
  (criterion: d-dim. integration of the forces?)

Algorithmic Problem:

- how to avoid $\mathcal{O}(N^2)$ complexity?
- key ideas: approximate long range forces, clustering of particles

Part I

The Barnes-Hut Algorithm

J. Barnes, P. Hut: *A hierarchical O(N log N) force-calculation algorithm*. Nature 324, 1986, p. 446 ff.

Barnes Hut Method – Key Ideas

Consider Astrophysics:

- force w.r.t. a far-away individual star might be neglected
- but not the force w.r.t. a far-away(?) galaxy
- thus: approximate forces on a individual star by grouping far-away stars, galaxies, etc. into clusters
- represent clusters by accumulated mass located at its centre-of-mass

Clustering via Domain Decomposition:

- clustering of particles required, where size of clusters depends on the distance to each individual particle
- solved by multi-level tree-based domain decomposition

Domain Decomposition

- distribute long-range region into subdomains: $\Omega^{\text{far}} = \bigcup_i \Omega_i^{\text{far}}$
- to be done for every particles position
  (in practice via hierarchical domain decomposition)
- assign a point y_0^i to each Ω_i^{far}
- decomposition depending on size of subdomains:

$$\text{diam} := \sup_{y \in \Omega_i^{\text{far}}} \|y - y_0^i\|$$

- choose decomposition such that

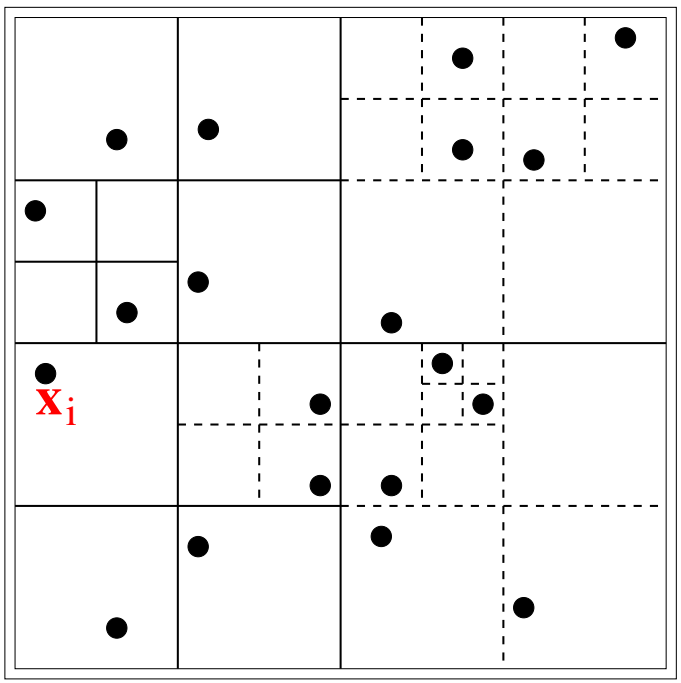
$$\frac{\text{diam}}{\|x - y_0^i\|} \leq \theta$$

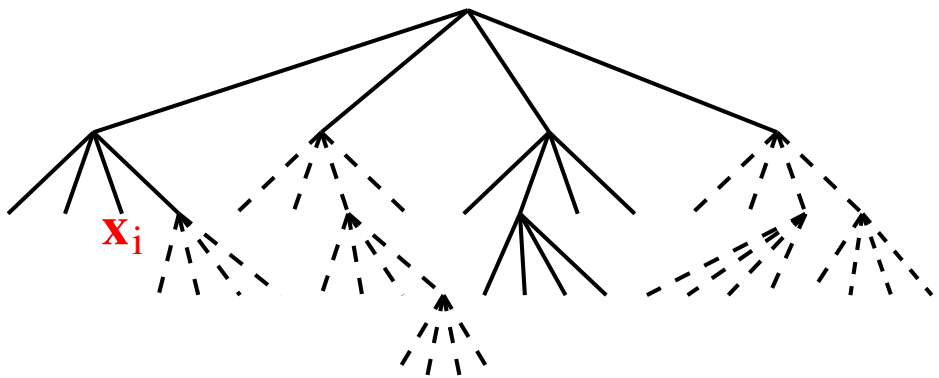
  for a suitable constant $0 < \theta < 1$

Octrees for Domain Decomposition

- efficient realisation of required decompositions
- recursive decomposition of Ω in subdomains
- stop, if only one particle left per cell
- use respective subtree for each x_i

Octrees:





Barnes-Hut Algorithm

- developed 1986 for applications in Astrophysics
- for gravity potential/force:

$$U(r_{ij}) = -\gamma_{\text{grav}} \frac{m_i m_j}{r_{ij}} \qquad \vec{F}_{ij} = \vec{F}(\vec{r}_i, \vec{r}_j) = -\gamma_{\text{grav}} \frac{m_i m_j (\vec{r}_i - \vec{r}_j)}{\|\vec{r}_i - \vec{r}_j\|^3}$$

- uses octree with 0 or 1 particles per cell
- inner nodes corresp. to clusters of particles (**pseudo particle**)
- idea: gravity force of particle cluster approximated (sum of masses, localised in centre of mass)
- computation of forces: for each particle, do an incomplete(!) octree traversal

Barnes-Hut: Computation of Forces

For each particle (position $x \in \Omega$):

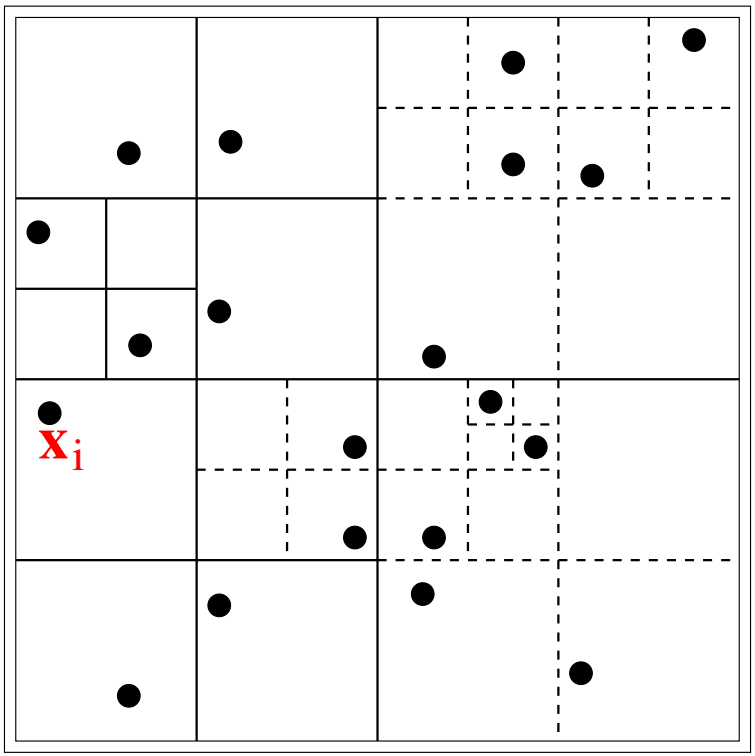
- start in root node
- descent into subdomains, until θ-rule satisfied: $\frac{diam}{r} \leq \theta$, r the distance of pseudo particle from x
- accumulate corresp. partial force to current particle

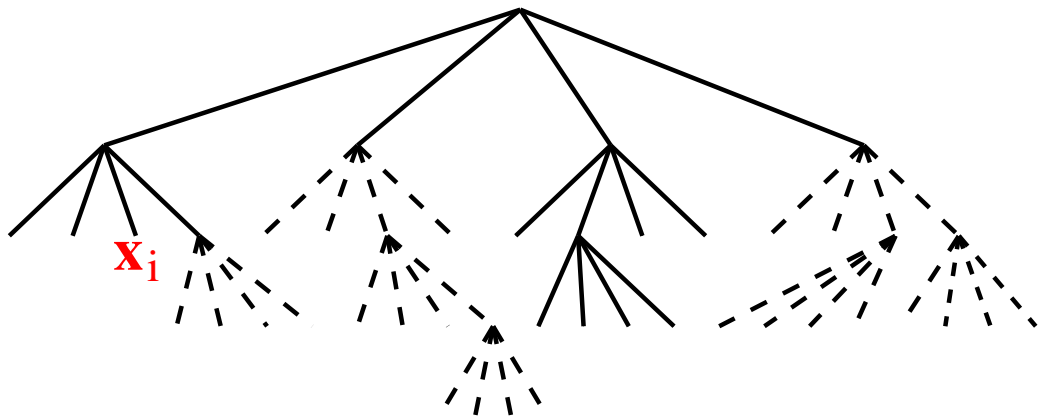
Implicit separation of short- and longe-range forces:

- short-range: all leaf nodes that are reached (containing 1 particle)
- long-range: all inner nodes, where descent is stopped (force caused by pseudo particle)

Barnes-Hut: Computation of Forces (2)

Tree traversal:





Barnes-Hut: Accuracy and Complexity

Accuracy of Barnes-Hut:

- depends on choice of θ
- the smaller θ, the more accurate the long-range forces
- the smaller θ, the larger the short-range (i.e., the costs)
- slow convergence w.r.t. θ (low-order method)

Complexity:

- grows for small θ
- for $\theta \to 0$: algorithm degenerates to "all-to-all" $\to \mathcal{O}(N^2)$
- for more or less homogeneously distributed particles:
  - number of active cells: $\mathcal{O}(\log N/\theta^3)$
  - total effort therefore $\mathcal{O}(\theta^{-3} N \log N)$

Barnes-Hut: Implementation

- computation of pseudo particles:
  - bottom-up-traversal (post-order)
  - sum up masses, weighted average for centre-of-mass
- computation of forces:
  - traversal of entire tree (outer loop on all particles)
  - top-down traversal (pre-order) until θ-rule satisfied (inner loop)
- further traversals for time integration
- re-build (or update) octree structure after each time step
  $\rightarrow$ requires efficient data structures and algorithms

Part II

The Fast Multipole Method

L. Greengard, V. Rokhlin: *A fast algorithm for particle simulations.* J. Comp. Phys. 73, 1987.
C. R. Anderson: *An implementation of the Fast Multipole Method without multipoles.*. SIAM J. Sci. Stat. Comput. 13(4), 1992.

Barnes-Hut vs. Fast Multipole

Barnes-Hut: compute forces of pseudo particles to particles:

Fast Multipole: compute forces between pseudo particles:

“Box–Box Interactions” in Hierachical Methods

What components are required:

- approximate potential of sets/clusters of particles
  $\rightarrow$ Barnes-Hut: pseudo particles in tree cells
  $\rightarrow$ Fast Multipole: higher-order representations
- hierarchical computation of “box potentials”
  $\rightarrow$ Barnes-Hut: combine pseudo particles in child cells
  $\rightarrow$ Fast Multipole: generate high order “box potentials”
- “box–box interactions” (too inaccurate for simple pseudo particles)
  $\rightarrow$ only in Fast Multipole:
  - potential/forces of a box onto all particles of a remote other box
  - potential/forces of accumulated box–box interactions of all particles (and child boxes!) within a box

Approximate “Box potentials”

Approximate the potential of a set/cluster of particles by:

- multipole extension (Greengard & Rokhlin, 1987)
  $\rightarrow$ similar concept as Taylor series
  $\rightarrow$ complicated to derive in 3D (spherical harmonics)
  $\rightarrow$ complicated formula for hierarchical assembly
- inner/outer ring approximations (Anderson, 1992)
  $\rightarrow$ derived via numerical integration of an integral formula
  $\rightarrow$ uniform interaction with child and remote boxes
  $\rightarrow$ hierarchical assembly via evaluation of potentials at integration points
- both approaches: principle of **“well-separated boxes”**
  $\rightarrow$ box–box interaction allowed between boxes that are separated by one box of the same hierarchical level

Outer Ring Approximations

- fundamental idea: represent potential via a surface integral

$$\Psi_a(\vec{x}) = \int_{S^2} g(a\vec{s}) \left(\sum_{n=0}^{\infty} \left(\frac{a}{r} \right)^{n+1} Q_n(\vec{s} \cdot \vec{x}_p) \right) ds$$

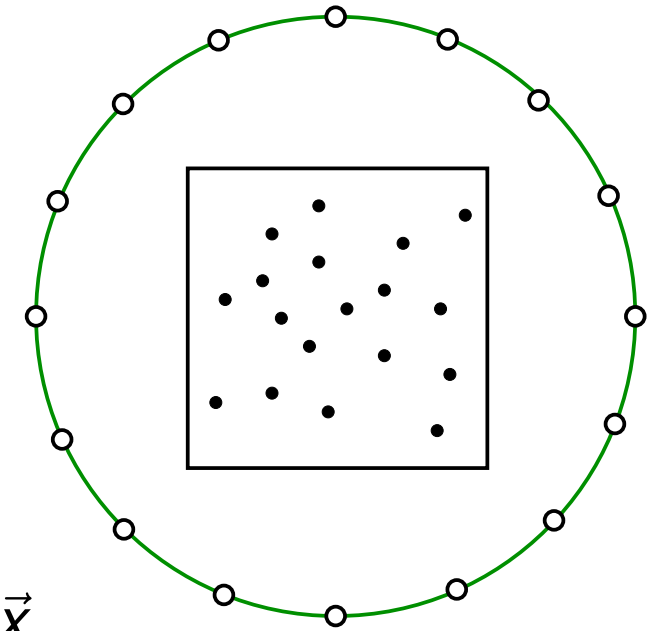
  in spherical coordinates $\vec{x} = (r, \theta, \psi)$, with suitable functions Q_n, $\vec{x}_p = (1, \theta, \psi)$, and $g(a\vec{s})$ the potential on a sphere of radius a that contains all particles
- use numerical integration rule:

$$\Psi_a(\vec{x}) \approx \sum_{i=1}^{K} w_i \, g(a\vec{s}_i) \left(\sum_{n=0}^{M} \left(\frac{a}{r} \right)^{n+1} Q_n(\vec{s}_i \cdot \vec{x}_p) \right)$$

  using K integration points $\vec{s}_i$ on the sphere S^2 (weights w_i) and choosing M relative to accuracy of integration rule

Outer Ring Approximations (2)

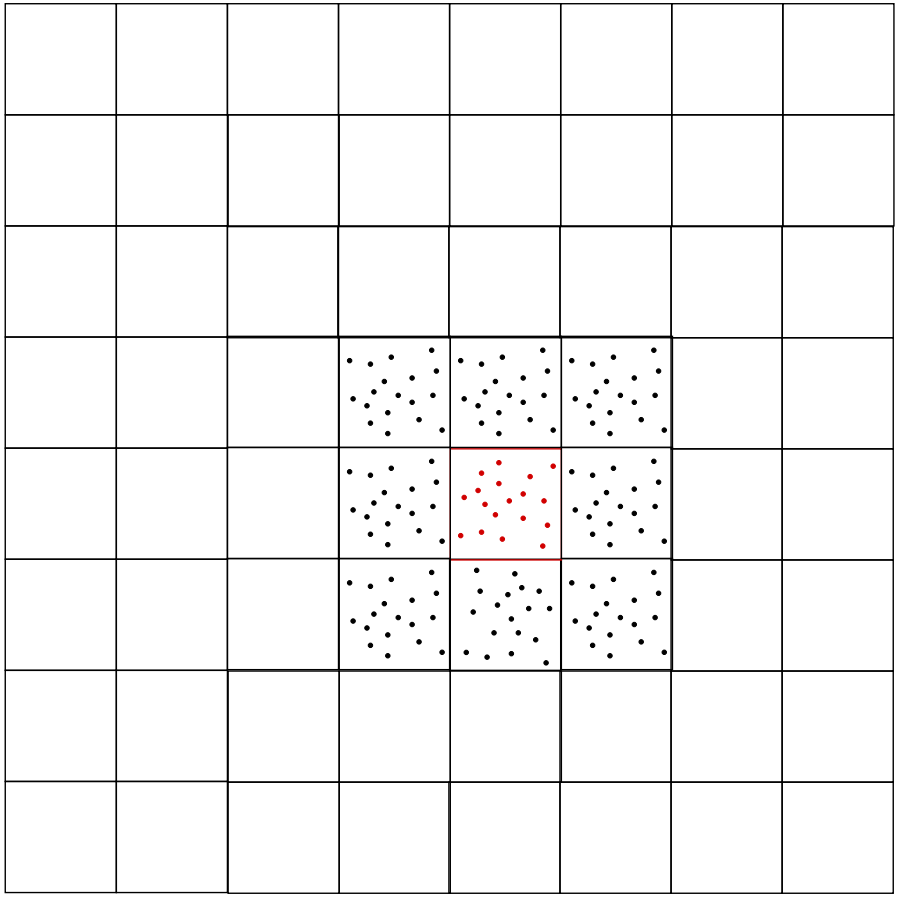
$$\Psi_a(\vec{x}) \approx \sum_{i=1}^{K} w_i \, g(a\vec{s}_i) \left(\sum_{n=0}^{M} \left(\frac{a}{r} \right)^{n+1} Q_n(\vec{s}_i \cdot \vec{x}_p) \right)$$

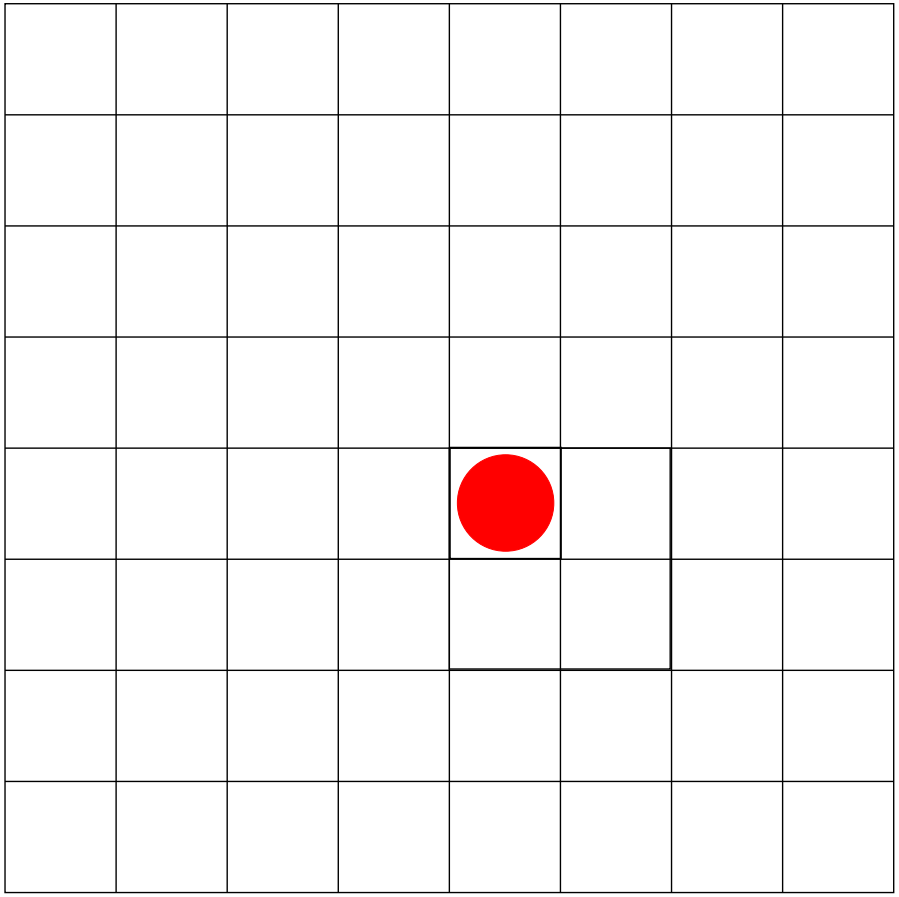


- w_i and $\vec{s}_i$ determined by integration rule
- r and $\vec{x}_p$ determined by evaluation position $\vec{x}$
- choice of a: usually twice the size of the box
- only $g(a\vec{s}_i)$ need to be computed and stored for each box:
  - $\rightarrow$ leaf box: accumulate potentials of all box-local particles at $a\vec{s}_i$
  - $\rightarrow$ parent box: accumulate approximate potentials of all child boxes at $a\vec{s}_i$

Hierarchical Force Computation – Illustrations

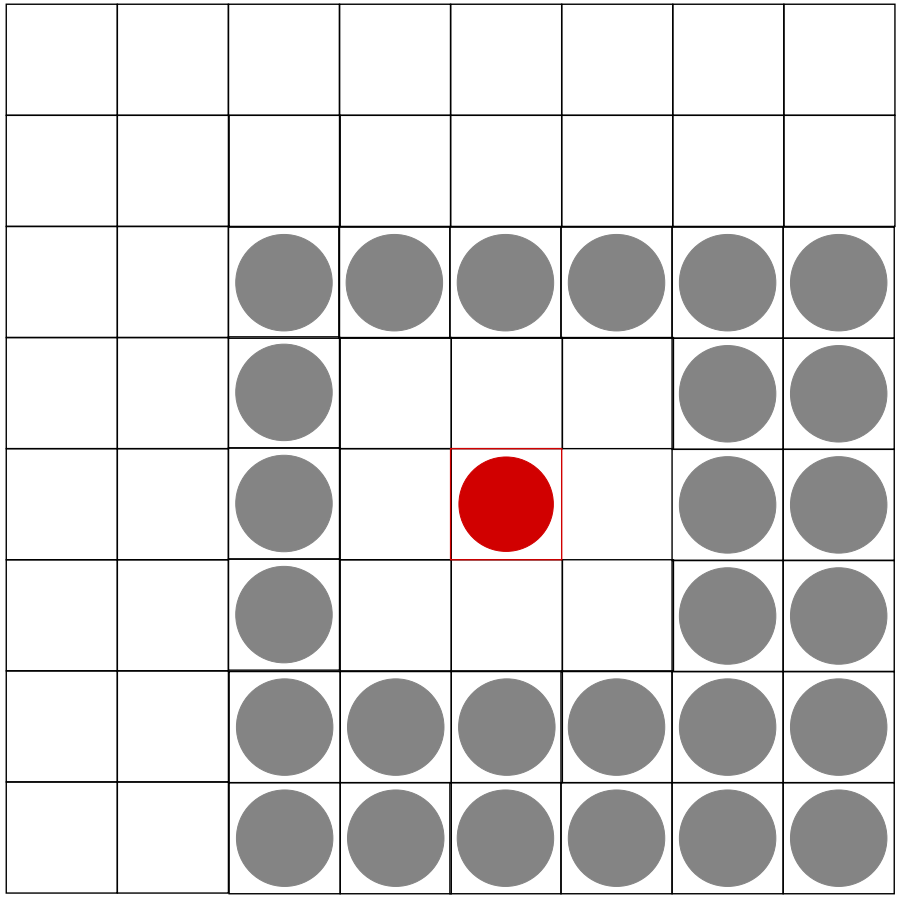
Forces on particles:

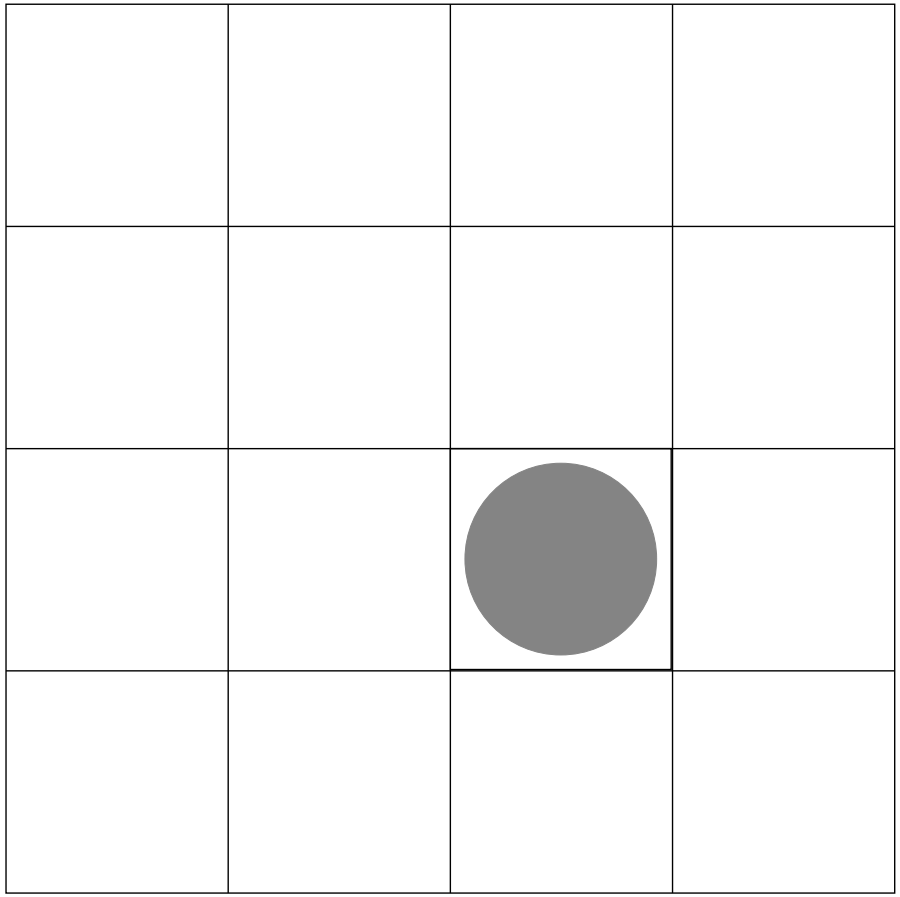




Hierarchical Force Computation – Illustrations

Forces on ring approximations:





Inner Ring Approximations

Almost the same formula:

$$\Psi_a(\vec{x}) \approx \sum_{i=1}^{K} w_i \, g(a\vec{s}_i) \left(\sum_{n=0}^{M} \left(\frac{r}{a} \right)^{n+1} Q_n(\vec{s}_i \cdot \vec{x}_p) \right)$$

- parent box: evaluate $\Psi_a(\vec{x})$ at all integration points of child-box rings
  $\Rightarrow$ obtain ring approximations of child boxes
- leaf box: evaluate $\Psi_a(\vec{x})$ at all interior particles
  $\Rightarrow$ obtain approximate box-induced forces on each particle

Hierarchical Force Computation

Forces on particles include:

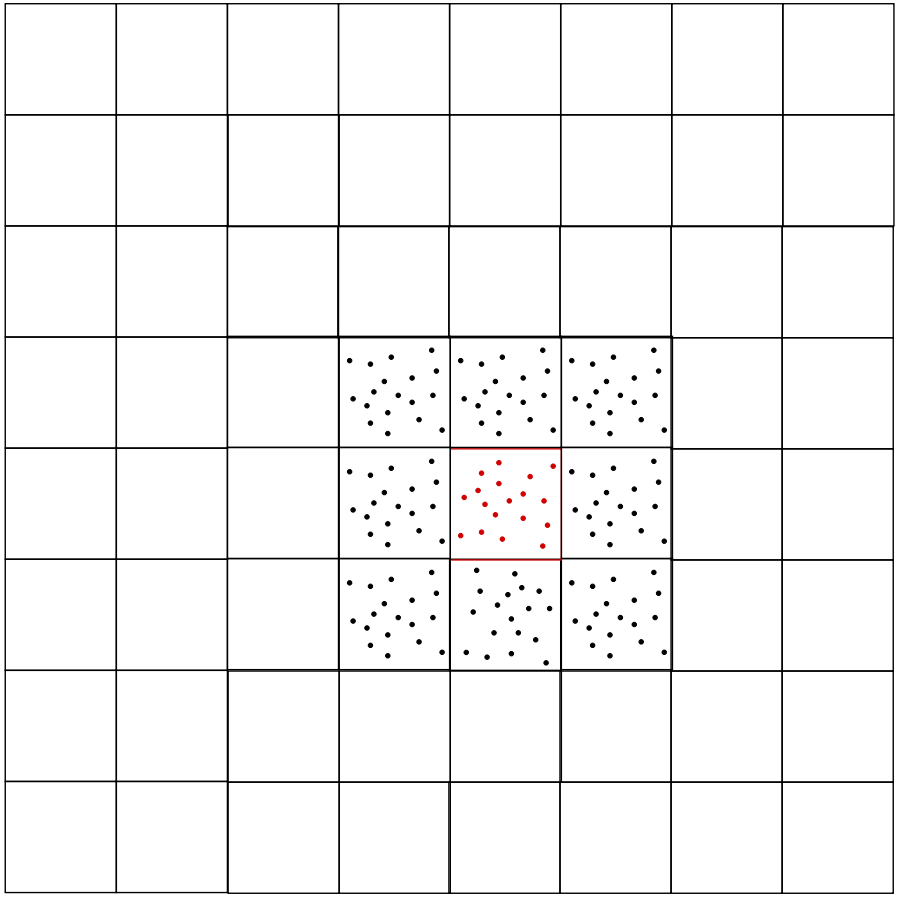
- direct interaction, if particles are in the same leaf-box or in a not well-separated leaf-box
- inner ring approximation with its containing box
  - $\rightarrow$ contains all accumulated box–box interactions with this box

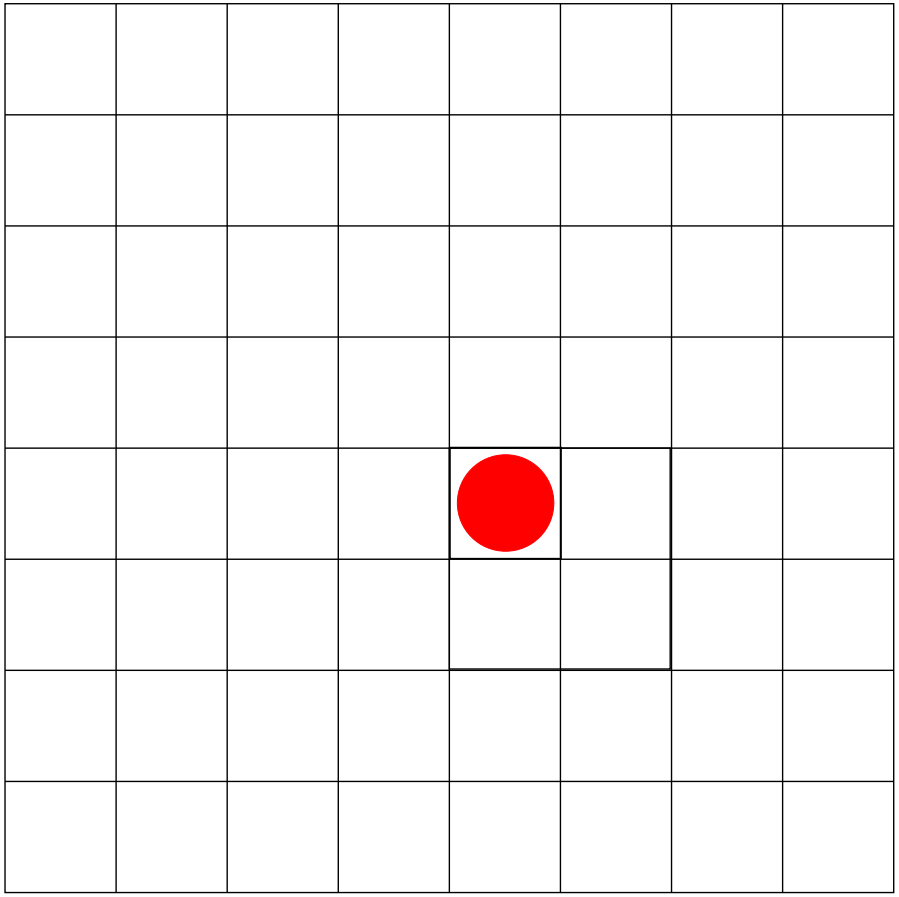
Forces on boxes include:

- box–box interaction with boxes that are not well-separated
- inner ring approximation with its parent box
  - $\rightarrow$ contains all accumulated box–box interactions with well-separated boxes

Hierarchical Force Computation – Illustrations

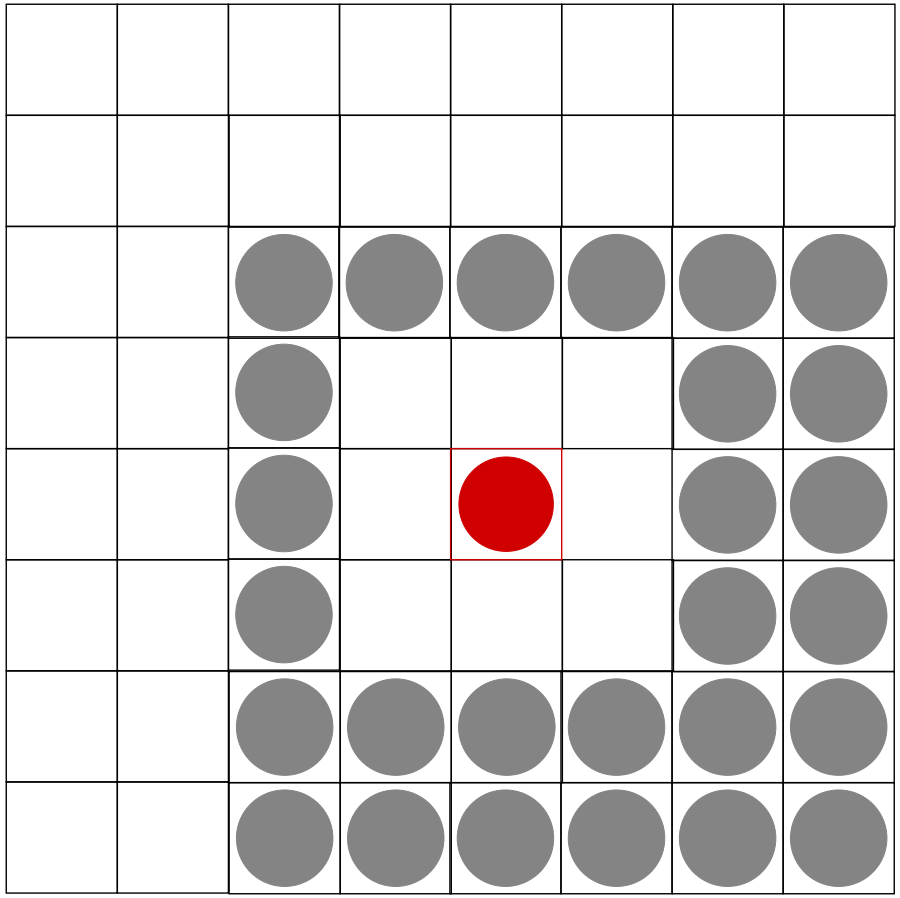
Forces on particles:

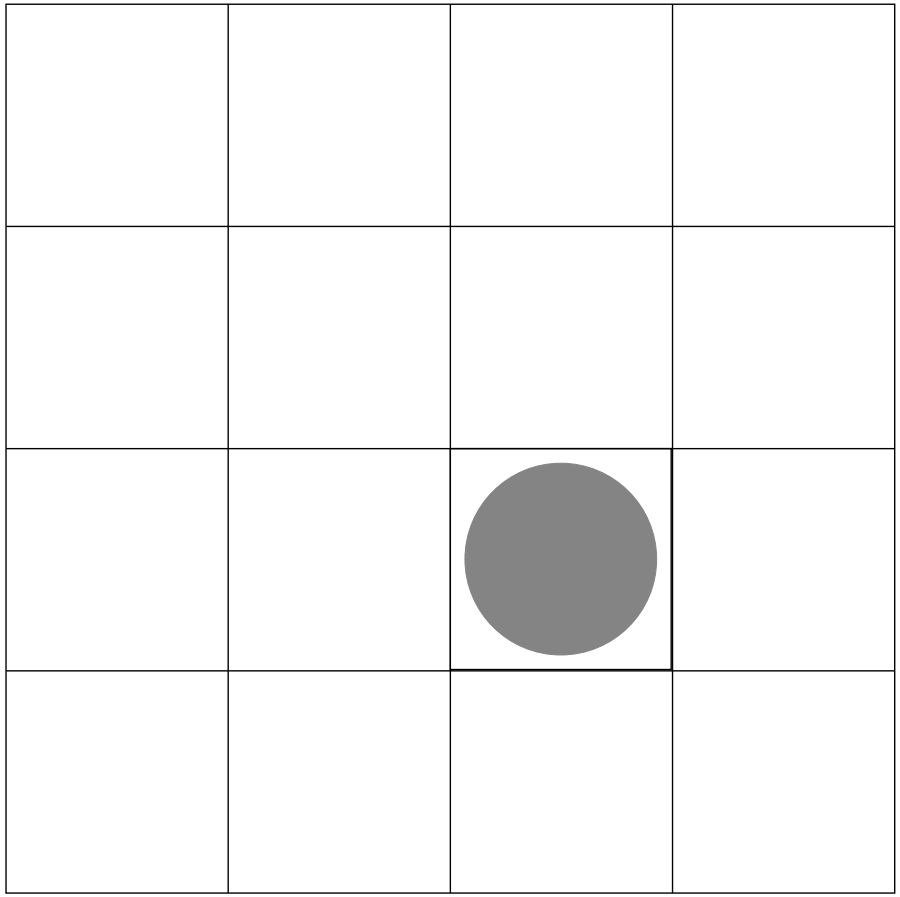




Hierarchical Force Computation – Illustrations

Forces on ring approximations:





Fast Multipole Method – Accuracy and Complexity

Accuracy:

- depends on accuracy of integration rule
  $\rightarrow$ determined by number of integration points
- in practice: can be increased to allow approximations that are accurate up to machine precision

Complexity:

- computation of box-approximations, i.e., all $g(a\vec{s}_i)$
  - $\rightarrow$ constant effort per box (leaf and inner boxes)
  - $\rightarrow$ thus $O(N_B)$ effort (N_B boxes); if max. number of particles per box is constant then $O(N)$ (N particles)
- computation of forces
  - $\rightarrow$ multilevel algorithms leads to $O(N)$ effort