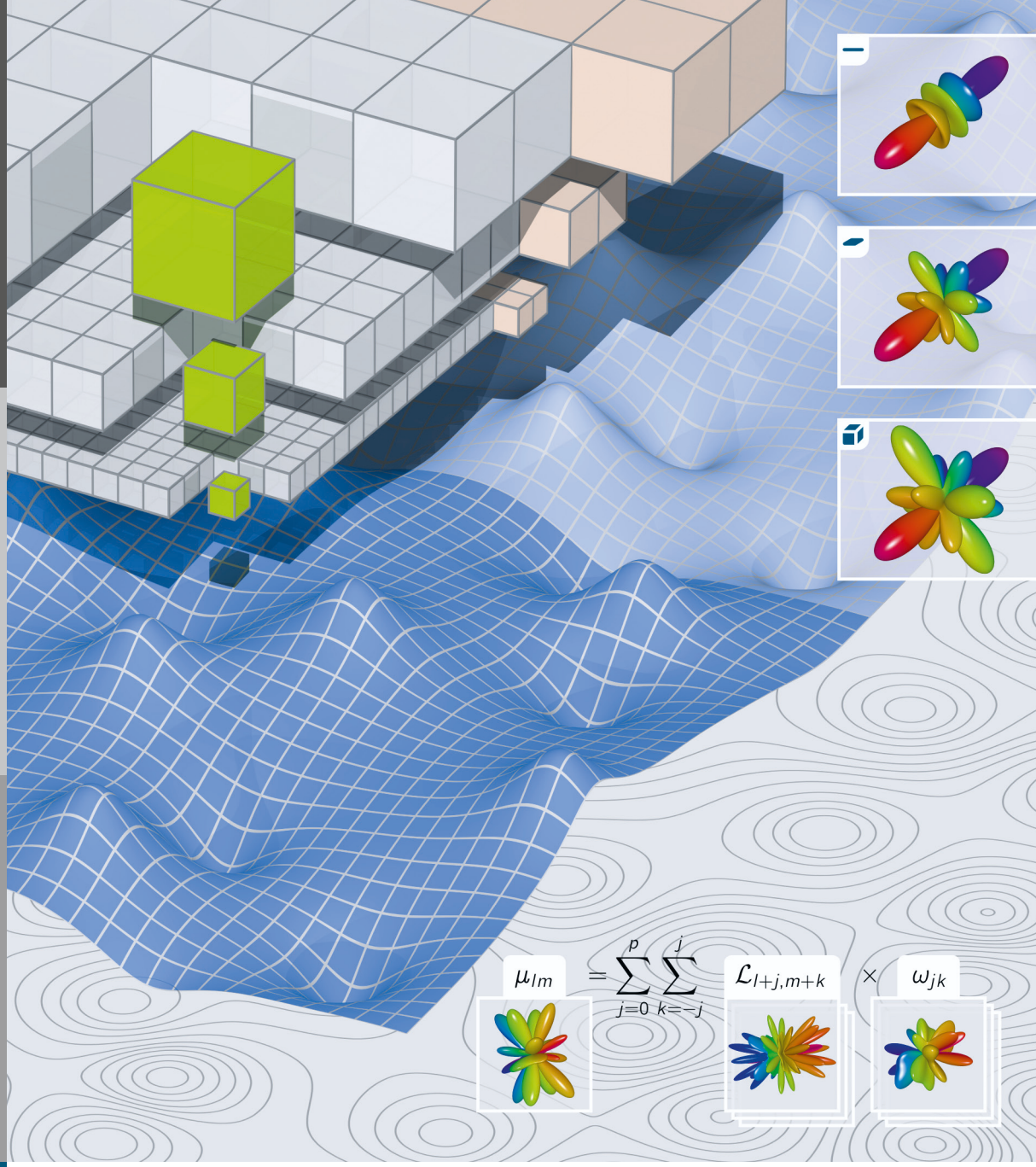


Periodic Boundary Conditions and the Error-Controlled Fast Multipole Method

Ivo Kabadshow

Member of the Helmholtz Association

JÜLICH
FORSCHUNGSZENTRUM

Schriften des Forschungszentrums Jülich
IAS Series Volume 11

Forschungszentrum Jülich GmbH
Institute for Advanced Simulation (IAS)
Jülich Supercomputing Centre (JSC)

Periodic Boundary Conditions and the Error-Controlled Fast Multipole Method

Ivo Kabadshow

Schriften des Forschungszentrums Jülich
IAS Series Volume 11

ISSN 1868-8489 ISBN 978-3-89336-770-2

Bibliographic information published by the Deutsche Nationalbibliothek.
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at http://dnb.d-nb.de.

Publisher and Distributor:	Forschungszentrum Jülich GmbH Zentralbibliothek 52425 Jülich Phone +49 (0) 24 61 61-53 68 · Fax +49 (0) 24 61 61-61 03 e-mail: zb-publikation@fz-juelich.de Internet: http://www.fz-juelich.de/zb
Cover Design:	Grafische Medien, Forschungszentrum Jülich GmbH
Source Cover Image:	Ivo Kabadshow
Printer:	Grafische Medien, Forschungszentrum Jülich GmbH
Copyright:	Forschungszentrum Jülich 2012

Schriften des Forschungszentrums Jülich
IAS Series Volume 11

D 468 (Diss., Wuppertal, Univ., 2010)

ISSN 1868-8489
ISBN 978-3-89336-770-2

The complete volume is freely available on the Internet on the Jülicher Open Access Server (JUWEL) at http://www.fz-juelich.de/zb/juwel

Persistent Identifier: urn:nbn:de:0001-2012020810
Resolving URL: http://www.persistent-identifier.de/?link=610

Abstract

The simulation of pairwise interactions in huge particle ensembles is a vital issue in scientific research. Especially the calculation of long-range interactions poses limitations to the system size, since these interactions scale quadratically with the number of particles. Fast summation techniques like the Fast Multipole Method (FMM) can help to reduce the complexity to $\mathcal{O}(N)$. This work extends the possible range of applications of the FMM to periodic systems in one, two and three dimensions with one unique approach. Together with a tight error control, this contribution enables the simulation of periodic particle systems for different applications without the need to know and tune the FMM specific parameters. The implemented error control scheme automatically optimizes the parameters to obtain an approximation for the minimal runtime for a given energy error bound.

Zusammenfassung

Die Simulation von paarweisen Wechselwirkungen für sehr große Teilchen-Ensembles ist eine wesentliche Aufgabe in der wissenschaftlichen Forschung. Besonders die Berechnung von langreichweitigen Wechselwirkungen legt der Systemgröße Beschränkungen auf, da die Anzahl der zu berechnenden Wechselwirkungen quadratisch mit der Anzahl der Teilchen skaliert. Schnelle Summationsverfahren wie die Fast Multipole Method (FMM) können dazu beitragen die Komplexität auf $\mathcal{O}(N)$ zu reduzieren. Diese Arbeit erweitert den möglichen Anwendungsrahmen der FMM auf Systeme mit ein, zwei oder drei-dimensionaler Periodizität in einem einheitlichen Ansatz. Zusammen mit einer präzisen Fehlerkontrolle erlaubt der hier vorgestellte Ansatz, die Simulation von Teilchensystemen für verschiedenste Anwendungen durchzuführen, ohne die für die Simulation benötigten FMM Parameter kennen oder bestimmen zu müssen. Die implementierte Fehlerkontrolle ermittelt diese Parameter automatisch, um damit eine Näherung für die minimale Simulationsrechenzeit für eine vorgegebene Energie-Fehlerschranke zu erhalten.

Contents

List of Figures

List of Tables

1. Introduction

> A system being given of a number whatever of particles attracting one another mutually according to Newton's law, it is proposed, on the assumption that there never takes place an impact of two particles, to expand the coordinates of each particle in a series proceeding according to some known function of time and converging uniformly for any space of time.
>
> *Gösta Mittag-Leffler, 1885 [1]*

1.1. The N-body Problem

This introducing quote was first printed in *Nature* over 120 years ago. Advised by Prof. Mittag-Leffler, King Oscar II of Sweden established a prize for the first one who could find a general solution to this specific problem later known as N-body problem. This challenge describes the interaction of N particles e.g. charges or masses, through e.g. the Coulomb force or the force of gravity.

Unfortunately, the problem could not be solved, so the prize had to be awarded to another important contributor to classical mechanics which was considered of great interest for the progress of science in those days. Finally Poincaré won the prize, even though he solved a different problem. Later on the originally stated problem was solved partly for three particles ($N = 3$) by Sundman [2]. In 1991 Qui-Dong developed a transformation leading to the global solution of the problem. Unfortunately, the speed of convergence is "terribly slow" [3]. Hence, for $N > 3$ no general solution was proposed until today. However, such systems can be solved numerically with the help of computers. Therefore, iterative methods are used to calculate forces at discrete time intervals on each particle prior to computing the corresponding velocities and updated positions.

Time Step loop of a N-body problem

A simple scheme using an iterative method to obtain all trajectories of N particles mutually interacting with each other is shown below. Since there is no global solution at hand, the algorithm represents an approximate solution to the problem. The first source of error is introduced by the finite number of digits available on a computer system, producing only approximate particle-particle forces. The second approximation occurs in the integration step. The time counter Δt cannot be decreased arbitrarily low due to the limited precision of the computer.

The force $\mathbf{F}_{ij}$ of particle j with charge (or mass) c_j acting on particle i with with charge (or mass) c_i is defined by

$$\mathbf{F}_{ij} = \frac{c_i c_j}{r_{ij}^3}\mathbf{r}_{ij} \ . \tag{1.1}$$

The distance between the two particles is denoted as r_{ij}. The following algorithm describes how a timestep loop of N mutually interacting particles can be computed.

1. Clear forces;

   for $i = 1$ *to* N **do**
   $\mathbf{F}_i = 0$
   end

2. Compute and accumulate forces;

   for $i = 1$ *to* $N - 1$ **do**
   for $j = i + 1$ *to* N **do**
   Compute force $\mathbf{F}_{ij}$ of particle j on particle i;
   $\mathbf{F}_i := \mathbf{F}_i + \mathbf{F}_{ij}$;
   $\mathbf{F}_j := \mathbf{F}_j - \mathbf{F}_{ij}$;
   end
   end

3. Integrate equations of motions;

   for $i = 1$ *to* N **do**
   $\mathbf{v}_i^{\text{new}} := \mathbf{v}_i^{\text{old}} + \frac{\mathbf{F}_i}{m_i}\Delta t$;
   $\mathbf{x}_i^{\text{new}} := \mathbf{x}_i^{\text{old}} + \mathbf{v}_i \Delta t$;
   end

4. Update time counter;

   $t := t + \Delta t$;

5. To calculate the subsequent timestep go back to step 1.

The algorithm [4] depicted above has a substantial computational overhead. Since the calculation of the double sum is represented by two nested `for` loops with each sum running up to the total number of particles N, the algorithm has a complexity of $\mathcal{O}(N^2)$. Doubling the number of particles results in an increase of runtime of a factor of four. For few particles this behavior may be negligible, but for a large number of particles this clearly slows down the simulation.

1.1.1. Simple Schemes to Reduce the Complexity

To reduce the overall complexity it is possible to drop interactions which do not have a relevant impact on the actual forces. Therefore, a cut-off radius is introduced. All particles within the cut-off are taken into account for interaction. All particles outside the sphere are neglected. The runtime and precision of the simulation clearly depends on the size of the sphere and hence the cut-off radius. Assuming a homogeneous particle system (constant particle density), the complexity can be reduced from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$, since the number of neighboring and hence interacting particles is constant for a certain cut-off.

1.1.2. Different Types of Potentials

We distinguish between two types of potentials depending on the value of the integral

$$I = \int \frac{1}{r^p} \, \mathrm{d}\mathbf{r} \, .$$

Short-range potentials decay rapidly with increasing distance r and the integral converges. In $\mathbb{R}^n$, potentials decaying faster than $1/r^p$ with $p > n$ are called short-range potentials, e.g. the Lennard Jones potential with $p = 6$.

Long-range potentials decay slowly with increasing distance r and the integral does not converge. In $\mathbb{R}^n$, potentials decaying slower than $1/r^p$ with $p \leq n$ are called long-range potentials. The Coulomb potential ($p = 1$) is a long-range potential decreasing as $1/r$.

Another classification of long- and short-range potentials from a physical point of view can be found in reference [5]. However, in all considerations the Coulomb potential $1/r$ is related to a long-range potential which does not allow a certain cut-off without violating the error bound.

1.2. Fast Summation Techniques

Unfortunately, cut-off schemes introduce undesirable artifacts [6] for long-range potentials. Therefore, one has to choose another path to reduce the complexity. The calculation of the forces in a Coulomb problem with kernel $K(x, y_i)$ and weights W_i can be generalized to a sum of the form

$$U(x) = \sum_{i=1}^{N} W_i K(x, y_i) \, .$$

Straightforward computation is considered to be a direct summation. The evaluation at N target points with N source points yields a $\mathcal{O}(N^2)$ complexity. Algorithms that reduce the cost of evaluating these sums at each of N target locations from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ or even $\mathcal{O}(N)$ will be referred to as fast summation schemes.

Historic Development of Fast Summation Algorithms

The first proposal was made by Ewald in 1921 for periodic systems in three dimensions. The overall complexity can be reduced to $\mathcal{O}(N^{3/2})$. With the advent of computers more sophisticated schemes emerged. In the 1960s Fedorenko [7] and Bakhvalov [8] first conceived a multigrid method, which was brought to an efficient utility by Brand [9] in 1977. The method – originally introduced to solve elliptic boundary-value problems – could be used for both open and periodic systems and has a complexity of order N. In 1980 Eastwood et al. [10] proposed a particle-mesh algorithm with a complexity of $\mathcal{O}(N \log N)$. This was followed by a treecode by Appel [11], Barnes and Hut [12]. Finally in 1987 Rokhlin and Greengard [13] proposed a method similar to the treecode method, but with *a priori* error bounds and linear complexity. This fast multipole method (FMM) was considered one of the top ten algorithms of the 20th century [14].

Basic Ideas Behind Fast Summation Schemes

The basic ideas behind fast summation schemes can be explained with a simple example. Consider the calculation of

$$U(x_j) = \sum_{i=1}^{N} W_i K(x_j, y_i) = \sum_{i=1}^{N} W_i (x_j - y_i)^2 \tag{1.2}$$

for target points x_j with $j = 1, \ldots, N$, where the points in the sets $\{x_j\}$ and $\{y_i\}$ are located on the real axis. Direct summation requires work proportional to $\mathcal{O}(N^2)$. We know

$$(x_j - y_i)^2 = x_j^2 - 2x_j y_i + y_i^2 \,.$$

Substitution into (1.2) leads to

$$U(x_j) = A_0 x_j^2 - 2A_1 x_j + A_2$$

where

$$A_k = \sum_{i=1}^{N} W_i y_i^k \,.$$

Precomputation of the moments A_k can be done in $\mathcal{O}(N)$, as well as the subsequent calculation of U at N target points. It is possible to generalize this scheme to summations (e.g. the Coulomb problem) where the kernel $K(x, y)$ can be expressed as a finite sum

$$K(x, y) = \sum_{k=1}^{p} \phi_k(x) \Psi_k(y) \,. \tag{1.3}$$

A fast summation algorithm is obtained by the following two–step procedure: First, compute the moments A_k defined by

$$A_k = \sum_{i=1}^{N} W_i \Psi_k(y_i) \,.$$

Second, evaluate $U(x)$ at each target position with

$$U(x) = \sum_{k=1}^{p} A_k \phi_k(x) \,.$$

The amount of work required is $\mathcal{O}(Np)$ rather than $\mathcal{O}(N^2)$. For $p \ll N$ the computation will be significantly faster than direct summation. A kernel that can be expressed as a finite sum as in (1.3), is called a degenerate or separable kernel. Independent of the details of the source distribution, $U(x)$ has to be a linear combination of the functions $\{\phi_1(x), \ldots, \phi_p(x)\}$. Therefore, the function space of $U(x)$ is tremendously smaller than the N-dimensional space containing the source and target data. Unfortunately, the kernel of interest for the Coulomb problem is not of this type. The singularity of the potential demands a more sophisticated scheme.

Key Features

In addition to the separable representation of the kernel, fast summation schemes dealing with potentials with a singularity e.g. Coulomb potential need to introduce a spatial decomposition in order to separate the singularities. Therefore, fast summation codes are based on a combinations of the following key features:

- A specified accuracy ε for the computation,
- A hierarchical spatial subdivision of sources,
- A far field expansion of the "kernel" $K(\mathbf{x}, \mathbf{y})$ in which the influence of source and evaluation points separates,
- (Optional) The conversion of far field expansions into local expansions.

2. FMM in Three Dimensions

After the emergence of fast summation methods, the numerical solution to a variety of different types of equations was accelerated. There exist methods accelerating the solution of the Helmholtz equation [15] as well as other (partial) differential equations [16]. However, when speaking about the FMM the most common implementation copes with the Laplace equation [17]. Besides, there exist various different schemes for one- [18, 19], two- [17], and three-dimensional [20] Laplace problems, since the computational overhead of the algorithm in one or two dimensions is significantly smaller.

Since our targeted application will be a simulation of particles with coordinates in three dimensions, the following FMM algorithm is derived for the Laplace equation in three dimensions. To simplify the expressions of the involved operators the derivation was performed in spherical rather than Cartesian coordinates. This chapter introduces the FMM for systems with open boundaries. The enhancements for periodic boundaries are added in chapter 3.

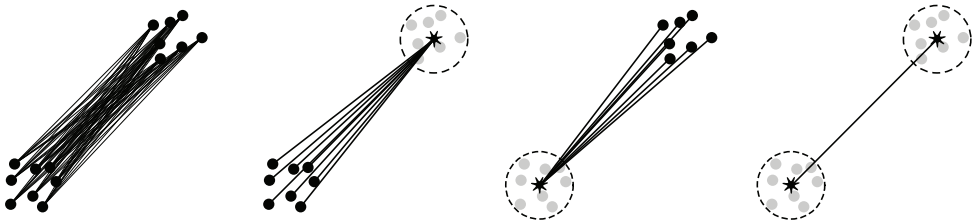
2.1. Informal Description of the FMM

The main idea behind fast summation methods, especially the FMM, is to give an approximate solution within a given precision goal ε of a certain quantity (energy, forces, potentials). However, this precision goal ε could be machine precision in which case the approximate solution does not differ from the "exact solution" when computed numerically. For regular potentials a fast summation scheme can be done by a simple factorization yielding the so called middleman scheme. Unfortunately, the potential we are dealing with has a singularity for $r \to 0$, thus the expansion is not valid everywhere. Therefore, we have to employ a different scheme involving a more complex algorithmic structure.

The presented approximation will benefit from the following observation. Assume a system of several clustered particles. The contribution on the force or energy, respectively from nearby particles on a test particle within a cluster will be dominant compared to the contribution from remote particles outside the cluster. Admittedly, the remote contribution is not zero. Neglecting these particles would correspond to a cut-off scheme without rigorous error control but ideal complexity $\mathcal{O}(N)$.

Grouping Source Particles

However, a remote particle from a spatial group will have almost the same influence on the test particle near the origin as another particle from the same remote group, since the distance



(a) 56 Direct Interactions (b) 8 Interactions: Grouped Source Particles (c) 7 Interactions: Grouped Target Particles (d) 1 Interaction: Grouped Source & Target Particles

Figure 2.1.: The figure depicts the main idea behind fast summation methods with respect to a particle system. The particles are distributed in space, but show clustering. Chart (a) shows the direct interaction of all particles in one cluster with all particles in the other cluster. Inter-Cluster interactions are not shown. Chart (b) shows the interaction via a source pseudo particle. Chart (c) shows the interaction via a target pseudo particle. Chart (d) shows the interaction with both, source and target pseudo particles.

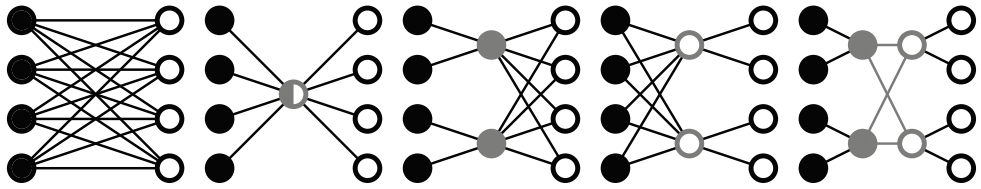
between the test particle at the origin and the remote particles is dominated by a large cluster–cluster distance. Therefore, all particles in a remote cluster could be combined together and may be represented by a new single pseudo particle with a new common center. The influence of several sources is combined into one source, which is depicted in figure 2.1b.

Grouping Target Particles

The grouping scheme can also be used in reverse. Considering a remote particle and a group of particles clustered together near the origin, the remote particle has almost the same influence on any source particle in the spatial group. Therefore, the interaction of this remote particle can be reduced to the interaction with a pseudo particle containing the clustered source particles (see figure 2.1c).

Grouping Both Source and Target Particles

Finally, the last two schemes can be combined together, allowing interactions of two pseudo particles, one at source and one at target particles. This scheme is depicted in figure 2.1d. Interaction takes place only via pseudo particles and not particles itself. This is advantageous since all particles in a simulation are sources and targets at the same time due to the mutual interaction. Hereby the number of interactions between the particles is reduced even more compared to asymmetric clustering of either source or target points.



(a) Direct (b) Middleman (c) Source Groups (d) Target Groups (e) Both Groups

Figure 2.2.: Schematic graphic of pairwise interactions of four particles. The black circles represent the sources, the open circles represent the targets. Sources and targets may be identical particles.

Defining Groups of Different Size

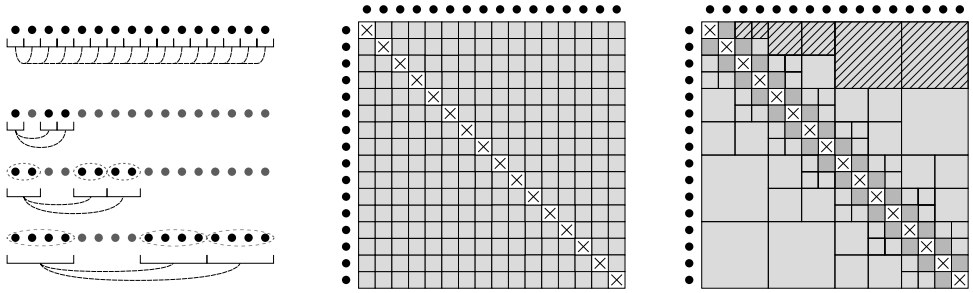
To define near and remote particles, a space decomposition scheme is used. For reason of simplicity we assume a recursive decomposition in cubic boxes. The different sized boxes are stored in a tree-like (oct-tree) data structure providing efficient construction of interaction sets, since source and target sets can be increased in size the more distant they are.

To generate the source and target sets from the given particle distribution, two operators are necessary. We distinguish between operators working vertically and horizontally inside the tree. First, the vertical operators shift expansions of source or target sets up and down the tree. Second, the horizontal operator translates source sets into target sets on each tree level.

However, it is possible that a limited number of nearby particles is not able to interact via pseudo particles. Therefore, these particles interact directly with each other. This restriction can be avoided by very deep trees, hence a very fine spatial decomposition where each particle has its own box. But this is not necessary, since the number of nearby particles is limited and thus does neither impair the overall complexity nor the computation time.

Interaction Sets

Figure 2.3 shows the interaction matrix of 16 particles. Every cell represents a single interaction pair. The crossed out cells represent interactions from particles with themselves which must not be calculated (singularity) and are dropped eventually. The first interaction matrix in figure 2.3b represents all $N(N-1)$ interactions. Due to symmetry, the lower left half contains the exact same elements of the upper half up to the sign. Thus, the overall interactions reduce to $\frac{1}{2}N(N-1)$. The second interaction matrix in figure 2.3c represents the interaction of the same particles via an FMM computation. Again, the crossed out cells are omitted. The dark-grayed cells represent direct neighbors, which have to be calculated directly. All other cells represent interactions via pseudo particles. The streaked cells are shown separately with their interaction sets in figure 2.3a. One can easily see that with increasing distance more and more source and target particles are grouped together which reduces the number of interactions dramatically. Again, the lower triangle is symmetric to the upper triangle similar to the direct summation.



(a) Direct and Clustered Interactions (b) Interaction Matrix (c) FMM Interaction Matrix

Figure 2.3.: Figure (a) depicts the direct interaction (first line) of a 1D system with 16 elements (black circles) and the interaction via the FMM scheme. Figure (b) depicts the interaction matrix of the direct interaction. Figure (c) depicts the interaction matrix via the FMM. Each square represents a certain particle-particle or multipole-multipole interaction.

To establish an accurate algorithm, providing the mentioned features, we need to introduce additional mathematical tools. In the next section we will derive the necessary theorems.

2.2. Mathematical Preliminaries

In three dimensions, functions which satisfy the Laplace equation

$$\nabla^2\Phi = \frac{\partial^2\Phi}{\partial x^2} + \frac{\partial^2\Phi}{\partial y^2} + \frac{\partial^2\Phi}{\partial z^2} = 0$$

are referred to as harmonic functions. The theory of such functions is called potential theory. A description of the theory can be found in [21, 22]. A solution satisfying the Laplace equation is e.g. $\Phi = 1/d$ with $d = \sqrt{(x-x_0)^2 + (y-y_0)^2 + (z-z_0)^2}$.

If a point particle of unit strength is fixed at $A = (x_0, y_0, z_0)$ then the potential due to this charge at an arbitrary but distinct point $R = (x, y, z)$ is given by

$$\Phi(R) = \frac{1}{d}$$

with d representing the distance between point R and A. The electrostatic field is given by

$$\vec{E}(R) = -\nabla\Phi = -\left(\frac{x-x_0}{d^3}, \frac{y-y_0}{d^3}, \frac{z-z_0}{d^3}\right).$$

Next we want to derive a series expansion for the potential at R in terms of the distance from the origin $\mathbf{r}$.

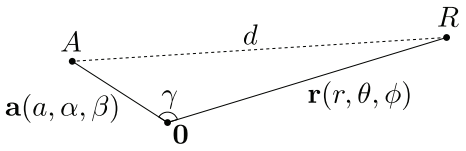


Figure 2.4.: Expansion of the inverse distance $1/d$ into the radial parts a and r and the subtended angle γ between $\overline{OA}$ and $\overline{OR}$.

2.2.1. Expansion of the Inverse Distance

Given two points $A(a, \alpha, \beta)$ and $R(r, \theta, \phi)$ with a, α, β and r, θ, ϕ being the spherical coordinates, we define the distance d as

$$d := |\mathbf{r} - \mathbf{a}| = \sqrt{r^2 + a^2 - 2ra\cos\gamma}\,.$$

Thus,

$$\frac{1}{d} = \frac{1}{r\sqrt{1 - 2\frac{a}{r}\cos\gamma + \frac{a^2}{r^2}}} = \frac{1}{r\sqrt{1 - 2u\mu + \mu^2}}\,,$$

having set

$$\mu = \frac{a}{r} \quad \text{and} \quad u = \cos\gamma\,.$$

For $\mu \leq 1$, the inverse square root can be expanded in powers of μ, resulting in the following series

$$\frac{1}{\sqrt{1 - 2u\mu + \mu^2}} = \sum_{l=0}^{\infty} P_l(u)\mu^l\,,$$

where

$$P_0(u) = 1, \quad P_1(u) = u, \quad P_2(u) = \frac{3}{2}(u^2 - \frac{1}{3}), \quad \ldots\,.$$

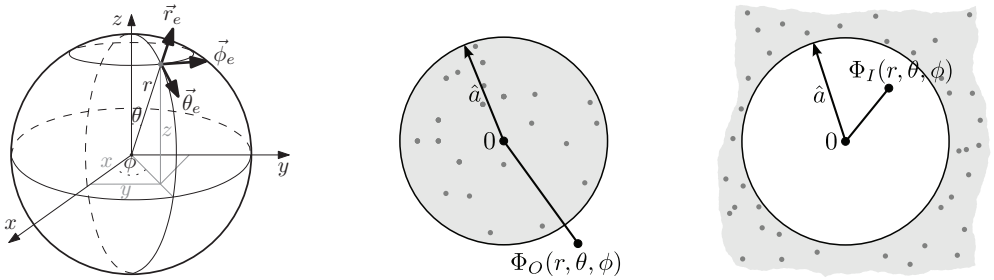
In general, the $P_l(u)$ are the Legendre polynomials [23] of degree l and are defined by

$$P_l(x) = \frac{1}{2^l l!}\frac{\mathrm{d}^l}{\mathrm{d}x^l}\left[(x^2 - 1)^l\right]\,.$$

Finally the expression for the inverse distance $1/d$ yields

$$\frac{1}{d} = \frac{1}{r}\sum_{l=0}^{\infty}\frac{a^l}{r^l}P_l(u) = \sum_{l=0}^{\infty}\frac{a^l}{r^{l+1}}P_l(\cos\gamma)\,. \tag{2.1}$$

The radial parts a and r of the two coordinates $\mathbf{a}$ and $\mathbf{r}$ are now factorized.



(a) Spherical Coordinate System

(b) Analytic domain of a multipole expansion of the potential Φ_O with $r > a$

(c) Analytic domain of a local expansion of the potential Φ_I with $r < a$

Figure 2.5.: The solution for the potential $\Phi(r, \theta, \phi)$ consists of two separate solutions. The domain can be split into two distinct parts. First, the local part inside the sphere. Second, the external part outside the sphere.

2.2.2. Spherical Harmonic Addition Theorem

Unfortunately, $P_l(\cos\gamma)$ does still depend on both coordinates A and R via $\cos\gamma$ and cannot be used to derive a fast summation scheme. A useful representation requires the introduction of spherical harmonics, allowing to factorize both source and target locations. By transforming the Laplace equation in spherical coordinates (see figure 2.5a), we get

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial\Phi}{\partial r}\right) + \frac{1}{r^2\sin\theta}\frac{\partial}{\partial\theta}\left(\sin\theta\frac{\partial\Phi}{\partial\theta}\right) + \frac{1}{r^2\sin^2\theta}\frac{\partial^2\Phi}{\partial\phi^2} = 0\,.$$

The solution of the equation can be found by assuming a separable solution of the form

$$\Phi(r,\theta,\phi) = R(r)T(\theta)P(\phi)$$

leading to an expression including spherical harmonics Y_{lm} and coefficients M_{lm} and L_{lm}

$$\Phi(r,\theta,\phi) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l} L_{lm}Y_{lm}(\theta,\phi)r^l + \frac{M_{lm}Y_{lm}(\theta,\phi)}{r^{l+1}}\,.$$

Remark 2.2.1. It should be noted that for a potential $\Phi_O(r,\theta,\phi)$ with $r > \hat{a}$, the coefficients L_{lm} have to be zero to satisfy the decay at infinity for the potential Φ_O as shown in figure 2.5b. For the potential $\Phi_I(r,\theta,\phi)$ with $r \leq \hat{a}$ inside the sphere with radius $\hat{a}$ the elements M_{lm} must be zero. (see figure 2.5c).

We define the spherical harmonics Y_{lm} and their complex-conjugate counterparts $Y^{\star}_{lm}$ as follows

$$Y_{lm}(\theta,\phi) = \sqrt{\frac{(2l+1)}{4\pi}}\sqrt{\frac{(l-m)!}{(l+m)!}}P_{lm}(\cos\theta)e^{im\phi}\,,$$
$$Y^{\star}_{lm}(\theta,\phi) = \sqrt{\frac{(2l+1)}{4\pi}}\sqrt{\frac{(l-m)!}{(l+m)!}}P_{lm}(\cos\theta)e^{-im\phi}\,.$$

The functions $P_{lm}(x)$ are called associated Legendre polynomials and can be derived as

$$P_{lm}(x) = (-1)^m(1-x^2)^{m/2}\frac{\mathrm{d}^m}{\mathrm{d}x^m}P_l(x)\,.$$

The number l is called *degree* and the number m is called *order* of the function $P_{lm}(x)$. The products

$$\cos(m\phi)P_{lm}(\cos\theta) \quad \text{and} \quad \sin(m\phi)P_{lm}(\cos\theta)$$

only depending on the spherical angles θ and ϕ are called *tesseral harmonics* for $m < l$ and *sectorial harmonics* for $m = l$. The special case for $m = 0$ with

$$P_{l0}(\cos\theta) = P_l(\cos\theta)$$

reduces associated Legendre polynomials to Legendre polynomials and are called *zonal harmonics*. The three different harmonics are shown in figures 2.6b, 2.6c and 2.6a.

To calculate the associated Legendre polynomials, recurrences can be used. For $m = l \geq 0$ we can define

$$P_{mm} = (-1)^m(2m-1)!!(1-x^2)^{m/2}\,,$$

where $n!!$ denotes the double factorial of n and can be defined for odd numbers n by

$$n!! = \begin{cases} 1, & \text{if} \quad n = -1 \\ n\cdot((n-2)!!) & \text{if} \quad n > -1\,. \end{cases}$$

For $l \neq m$ we obtain a recurrence relation

$$(l-m)P_{lm} = x(2l-1)P_{l-1,m} - (l+m-1)P_{l-2,m}\,.$$

Theorem 2.2.2. Addition theorem for Legendre polynomials *Let A and B be points with spherical coordinates (a,α,β) and (r,θ,ϕ) and let γ be the angle subtended between them. Then*

$$P_l(\cos\gamma) = \frac{4\pi}{2l+1}\sum_{m=-l}^{l} Y^{\star}_{lm}(\alpha,\beta)Y_{lm}(\theta,\phi)\,.$$

(a) Zonal Harmonic $Y_{10,0}$ (b) Tesseral Harmonic $Y_{10,5}$ (c) Sectorial Harmonic $Y_{10,10}$

Figure 2.6.: Real part of spherical harmonics with different degree and order.

The proof of the theorem can be found in a text by Rose [24]. It can be derived with the use of the Wigner D-matrices [25]. We will summarize this proof briefly. It is shown for arbitrary g with

$$g = \sum_{m=-l}^{l} Y_{lm}^{\star}(\alpha, \beta) Y_{lm}(\theta, \phi) \tag{2.2}$$

that g is invariant under rotations of the coordinate system with $\alpha \to \alpha', \beta \to \beta', \theta \to \theta', \phi \to \phi'$

$$g = \sum_{m=-l}^{l} Y_{lm}^{\star}(\alpha, \beta) Y_{lm}(\theta, \phi) = \sum_{m=-l}^{l} Y_{lm}^{\star}(\alpha', \beta') Y_{lm}(\theta', \phi') \,.$$

Therefore, g can be evaluated in any coordinate system. To obtain the addition theorem, we position point $A(a, \alpha, \beta)$ on the z-axis and point B in the xz-plane by setting $\theta = 0$. Now the xz-plane is defined as the plane containing the origin $\mathbf{0}$, A and B. Thus $\alpha = 0$ and $\phi = 0$, defining γ as angle subtended between the vectors of the two points. The angle-dependent part of point A reduces to

$$Y_{lm}(0, \beta) = \delta_{m0} \sqrt{\frac{2l+1}{4\pi}}$$

with δ_{ij} being the Kronecker delta, while for point B the spherical harmonic $Y_{lm}(\theta, \phi)$ reduces to $Y_{l0}(\gamma, 0)$. Equation (2.2) simplifies to

$$g = \sqrt{\frac{2l+1}{4\pi}} Y_{l0}(\gamma, 0) \,.$$

Since we can evaluate g in any coordinate system, we may write

$$\sqrt{\frac{2l+1}{4\pi}} Y_{l0}(\gamma, 0) = \sum_{m} Y_{lm}^{\star}(\alpha', \beta') Y_{lm}(\theta', \phi') \,.$$

By substituting and rearranging the prefactor

$$Y_{l0}(\gamma, 0) = \sqrt{\frac{2l+1}{4\pi}} P_l(\cos\gamma)\,,$$

we finally obtain the spherical harmonic addition theorem

$$P_l(\cos\gamma) = \frac{4\pi}{2l+1} \sum_{m=-l}^{l} Y_{lm}^{\star}(\alpha, \beta) Y_{lm}(\theta, \phi)\,.$$

Remark 2.2.3. By redefining the spherical harmonics Y_{lm} and $Y_{lm}^{\star}$ without the normalization factor $\sqrt{(2l+1)/4\pi}$ the addition theorem simplifies to

$$P_l(\cos\gamma) = \sum_{m=-l}^{l} Y_{lm}^{\star}(\alpha, \beta) Y_{lm}(\theta, \phi)\,.$$

With the help of the last equation, it is possible to separate the source (a, α, β) and target (r, θ, ϕ) coordinates from (2.1). The inverse distance $1/d$ is fully factorized by

$$\frac{1}{d} = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \frac{a^l}{r^{l+1}} Y_{lm}^{\star}(\alpha, \beta) Y_{lm}(\theta, \phi)\,.$$

2.2.3. Expansion of Particle-Particle Interactions

Following the notation of Head-Gordon [26] and the derived formulas from the last sections, we can expand the coordinates of a particle-particle interaction as follows

$$\begin{aligned}\frac{1}{|\mathbf{r}-\mathbf{a}|} &= \sum_{l=0}^{\infty} P_l(\cos\gamma) \frac{a^l}{r^{l+1}} \\ &= \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \frac{(l-m)!}{(l+m)!} \frac{a^l}{r^{l+1}} P_{lm}(\cos\alpha) P_{lm}(\cos\theta) e^{-im(\beta-\phi)}\,.\end{aligned}$$

We define the scaled associated Legendre polynomials $\tilde{P}_{lm}$ and $\tilde{\tilde{P}}_{lm}$ to ensure numerical stability and to simplify the FMM operators with

$$\tilde{P}_{lm} = \frac{1}{(l+m)!} P_{lm} \quad \text{and} \quad \tilde{\tilde{P}}_{lm} = (l-m)! P_{lm}\,.$$

Now, we define the multipole moments $\omega_{lm}^j(q_j, \mathbf{a}_j)$ for a particle at $\mathbf{a}_j$ with strength q_j about the origin $(0, 0, 0)$. The chargeless version of the multipole is defined by $O_{lm}^j(\mathbf{a}_j)$ with

$$\omega_{lm}^j(q_j, \mathbf{a}_j) = q_j O_{lm}^j(\mathbf{a}_j) = q_j a_j^l \tilde{P}_{lm}(\cos\alpha_j) e^{-im\beta_j}\,.$$

Multipole moments of multiple particles $j \in \{1, \ldots, k\}$ about a common origin can be summed, yielding

$$\omega_{lm}(q, \mathbf{a}) = \sum_{j=1}^{k} \omega_{lm}^{j}(q_j, \mathbf{a}_j) = \sum_{j=1}^{k} q_j O_{lm}^{j}(\mathbf{a}_j) = \sum_{j=1}^{k} q_j a_j^l \tilde{P}_{lm}(\cos\alpha_j) e^{-im\beta_j} .$$

We can also establish the coefficients of a local Taylor-like expansion of the potential at the origin due to a distant particle at $\mathbf{r}_j$. The chargeless version of the local Taylor-like expansion is defined by $M_{lm}^{j}(\mathbf{r}_j)$ as

$$\mu_{lm}^{j}(q_j, \mathbf{r}_j) = q_j M_{lm}^{j}(\mathbf{r}_j) = q_j \frac{1}{r_j^{l+1}} \tilde{\tilde{P}}_{lm}(\cos\theta_j) e^{im\phi_j} .$$

Again, coefficients with a common origin can be summed

$$\mu_{lm}(q, \mathbf{r}) = \sum_{j=1}^{k} \mu_{lm}^{j}(q_j, \mathbf{r}_j) = \sum_{j=1}^{k} q_j M_{lm}^{j}(\mathbf{r}_j) = \sum_{j=1}^{k} q_j \frac{1}{r_j^{l+1}} \tilde{\tilde{P}}_{lm}(\cos\theta_j) e^{im\phi_j} .$$

We denote chargeless multipole expansions O_{lm} and chargeless Taylor-like expansions M_{lm} for $k = 1$ without the superscript j, subsequently. The corresponding potential $\Phi(P)$ due to a set of particles can be defined via the following two theorems.

Theorem 2.2.4. Multipole expansion *Suppose that k particles of strengths $q_j, j = 1, \ldots, k$ are located at the points $\mathbf{a}_j = (a_j, \alpha_j, \beta_j)$, $j = 1, \ldots, k$ with $|a_j| < \hat{a}$ inside a sphere. Then for any $P = (r, \theta, \phi) \in \mathbb{R}^3$ with $r > \hat{a}$, the potential $\Phi(P)$ is given by*

$$\begin{aligned} \Phi(P) &= \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \omega_{lm}(q, \mathbf{a}) \frac{1}{r^{l+1}} \tilde{\tilde{P}}_{lm}(\cos\theta) e^{im\phi} \\ &= \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \omega_{lm}(q, \mathbf{a}) M_{lm}(\mathbf{r}) . \end{aligned}$$

Theorem 2.2.5. Local expansion (Taylor-like) *Suppose that k particles of strengths $q_j, j = 1, \ldots, k$ are located at the points $R_j = (r_j, \theta_j, \phi_j)$, $j = 1, \ldots, k$ outside the sphere with radius $\hat{a}$ with $\hat{a} < r_j$. Then for any $P = (a, \alpha, \beta) \in \mathbb{R}^3$ with $a < \hat{a}$, the potential $\Phi(P)$ is given by*

$$\begin{aligned} \Phi(P) &= \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \mu_{lm}(q, \mathbf{r}) a^l \tilde{P}_{lm}(\cos\alpha) e^{-im\beta} \\ &= \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \mu_{lm}(q, \mathbf{r}) O_{lm}(\mathbf{a}) . \end{aligned}$$

2.2.4. Addition Theorem for Regular Solid Harmonics

The addition theorem described in section 2.2.2 links a Legendre polynomial to a finite sum of associated Legendre polynomials. To establish the FMM algorithm, another addition theorem is needed, involving associated Legendre polynomials only. With the help of this recurrence relation it is possible to represent two associated Legendre polynomials with different centers by just one polynomial at a new center. A short proof of this theorem can be found in [27]. We derive the formula briefly. To this end, we need to introduce the solid harmonics given by

$$\begin{aligned} R_{lm}(\mathbf{r}) &= r^l P_{lm}(\cos\theta) e^{im\phi} \\ I_{lm}(\mathbf{r}) &= r^{-l-1} P_{lm}(\cos\theta) e^{im\phi}\,. \end{aligned}$$

The regular solid harmonics R_{lm} and the irregular solid harmonics I_{lm} are both solutions of the Laplace equation.

Remark 2.2.6. It should be noted at this point that the regular solid harmonics R_{lm} differ from the recently introduced O_{lm} only by a prefactor

$$R_{lm}(\mathbf{r}) = (l+m)!\, O^{\star}_{lm}(\mathbf{r})\,. \tag{2.3}$$

Remark 2.2.7. Similarly to the regular solid harmonics R_{lm}, the irregular solid harmonics I_{lm} can be written in terms of the elements M_{lm}

$$I_{lm}(\mathbf{r}) = \frac{1}{(l-m)!} M_{lm}(\mathbf{r})\,. \tag{2.4}$$

Let us express the vector $\mathbf{r}$ with its polar coordinates (r, θ, ϕ) in Cartesian coordinates (x, y, z)

$$x = r\sin\theta\cos\phi\,, \qquad y = r\sin\theta\sin\phi\,, \qquad z = r\cos\theta\,.$$

The three dimensional representation can be reduced to a two dimensional one by choosing $\phi = 0$ and expressing vector $\mathbf{r}$ in the complex plane. Hobson [28] showed that the generating functions of the solid harmonics can be written as

$$\begin{aligned} (z+ix)^l &= \sum_{m=-l}^{l} i^m \frac{l!}{(l+m)!} R_{lm}(\mathbf{r}) \\ (z+ix)^{-l-1} &= \sum_{m=-l}^{l} i^{-m} \frac{(l-m)!}{l!} I_{lm}(\mathbf{r})\,. \end{aligned}$$

To obtain the addition theorem for $\mathbf{r} = \mathbf{r}_1 + \mathbf{r}_2$ we know

$$(z+ix)^l = [(z_1+ix_1) + (z_2+ix_2)]^l$$

which yields after the expansion via the binomial theorem

$$(z+ix)^l = \sum_{j=0}^{l} \binom{l}{j} (z_1+ix_1)^j (z_2+ix_2)^{l-j}$$

with

$$\binom{l}{j} = \frac{l!}{j!(l-j)!}\,.$$

Substituting the generating function in the last equation yields

$$\sum_{m=-l}^{l} i^m \frac{l!}{(l+m)!} R_{lm}(\mathbf{r}) =$$
$$\sum_{j=0}^{l} \sum_{k_1=-j}^{j} \sum_{k_2=-(l-j)}^{l-j} \binom{l}{j} i^{k_1+k_2} \frac{j!}{(j+k_1)!} \frac{(l-j)!}{(l-j+k_2)!} R_{j,k_1}(\mathbf{r}_1) R_{l-j,k_2}(\mathbf{r}_2)\,.$$

After identifying the coefficients of terms with equal numbers $m = k_1 + k_2$ and $k_1 = k$ and $k_2 = m - k$ it follows

$$\sum_{m=-l}^{l} i^m \frac{l!}{(l+m)!} R_{lm}(\mathbf{r}) =$$
$$\sum_{m=-l}^{l} \sum_{j=0}^{l} \sum_{k=-j}^{j} i^m \frac{l!}{(j+k)!(l-j+m-k)!} R_{j,k}(\mathbf{r}_1) R_{l-j,m-k}(\mathbf{r}_2)\,.$$

Dropping the outer sum and rearranging the terms the remaining equation simplifies to

$$R_{lm}(\mathbf{r}_1+\mathbf{r}_2) = \sum_{j=0}^{l} \sum_{k=-j}^{j} \binom{l+m}{j+k} R_{j,k}(\mathbf{r}_1) R_{l-j,m-k}(\mathbf{r}_2)\,. \tag{2.5}$$

Equation (2.5) defines the addition theorem for regular solid harmonics. The addition theorem for irregular harmonics can be derived similarly.

2.2.5. Addition Theorem for Irregular Solid Harmonics

For irregular solid harmonics I_{lm} we have to assume $|z_1+ix_1| > |z_2+ix_2|$ to allow a binomial expansion for $\mathbf{r} = \mathbf{r}_1 - \mathbf{r}_2$ in the following equation

$$(z+ix)^{-l-1} = [(z_1+ix_1) - (z_2+ix_2)]^{-l-1}\,.$$

The expansion yields

$$(z+ix)^{-l-1} = \sum_{j=0}^{\infty}(-1)^j \binom{-l-1}{j}(z_2+ix_2)^j(z_1+ix_1)^{-j-l-1}\,.$$

With the help of the generalized binomial coefficients

$$\binom{l}{j} = (-1)^j \binom{-l+j-1}{j}$$

and the substitution of the solid harmonics with their generating functions $R_{lm}(\mathbf{r})$ and $I_{lm}(\mathbf{r})$ we get

$$\sum_{m=-l}^{l} i^{-m}\frac{(l-m)!}{l!} I_{lm}(\mathbf{r}_1-\mathbf{r}_2) =$$
$$\sum_{j=0}^{\infty}\sum_{k_1=-l-j}^{l+j}\sum_{k_2=-j}^{j}\binom{l+j}{j} i^{k_2-k_1}\frac{j!(l+j-k_1)!}{(j+k_2)!(l+j)!} R_{j,k_2}(\mathbf{r}_2) I_{l+j,k_1}(\mathbf{r}_1)\,.$$

Again, identifying coefficients with $m = k_1 + k_2$ and $k_2 = -k$ and $k_1 = m + k$ it follows

$$\sum_{m=-l}^{l} i^{-m}\frac{(l-m)!}{l!} I_{lm}(\mathbf{r}_1-\mathbf{r}_2) =$$
$$\sum_{m=-l}^{l}\sum_{j=0}^{\infty}\sum_{k=-j}^{j}\frac{1}{l!} i^{-m-2k}\frac{(l+j-m-k)!}{(j+k)!}(-1)^k R^{\star}_{jk}(\mathbf{r}_2) I_{j+l,m+k}(\mathbf{r}_1)\,.$$

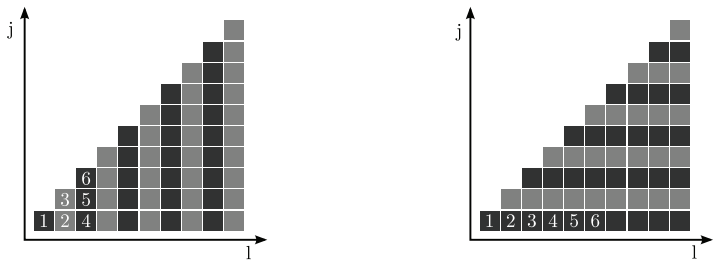
For simplicity and to avoid an additional $(-1)^k$ we used the complex conjugate of R_{jk}. After dropping the outer summation over m we identify the addition theorem for $I_{lm}(\mathbf{r}_1-\mathbf{r}_2)$ as

$$I_{lm}(\mathbf{r}_1-\mathbf{r}_2) = \sum_{j=0}^{\infty}\sum_{k=-j}^{j}\frac{(l+j-m-k)!}{(j+k)!(l-m)!} R^{\star}_{jk}(\mathbf{r}_2) I_{j+l,m+k}(\mathbf{r}_1)\,. \tag{2.6}$$

2.2.6. Formal Double Sum Manipulations

To obtain the FMM operators several infinite sums have to be manipulated with respect to the ordering of the elements. Following Choi [29] for absolutely convergent sums we have

$$\sum_{j=n}^{\infty}\sum_{l=j+(p-1)n}^{\infty} A_{jl} = \sum_{l=pn}^{\infty}\sum_{j=n}^{l-(p-1)n} A_{jl} \quad \text{with} \quad p-1\in\mathbb{N}; n+1\in\mathbb{N}\,.$$



(a) Summation over j and l (b) Summation over l and j

Figure 2.7.: The gray and black cells represent a certain element A_{jk} in the summation order. Figure (a) shows the inner summation along the j axis followed by the summation over l. Figure (b) shows the summation order, starting with the summation along the l axis [30] followed by the summation over j. For absolute convergent sums the summation order can be interchanged.

To derive our FMM operators we need a simpler form of the identity with $p = 2$ and $n = 0$, which corresponds to the summation over the infinite triangle $0 \leq j \leq l$ in two different ways as depicted in figure 2.7.

$$\sum_{j=0}^{\infty}\sum_{l=j}^{\infty} A_{jl} = \sum_{l=0}^{\infty}\sum_{j=0}^{l} A_{jl} \tag{2.7}$$

with $\{(j,l) : l \geq j, j \in \mathbb{N}\}$ for the left double sum and $\{(j,l) : j \leq l, l \in \mathbb{N}\}$ for the right double sum representing the same elements A_{jl}.

2.3. Mathematical Operators

In this section we describe the required FMM operators, which can be derived from the preliminaries of the last section to obtain a fast summation scheme. We need three different operators to establish the FMM scheme. Two operators for the vertical up- and down-shifts between the different tree levels and one operator for the conversion of remote multipole expansions at each tree level. We denote the truncation of an expansion with p.

2.3.1. Translation of a Multipole Expansion (M2M)

With help of the addition theorem in (2.5) we are now able to shift the coefficients of a multipole expansion around a point located at $\mathbf{a}$ to a point located at $\mathbf{a} + \mathbf{b}$ (Fig 2.8). Since

$$R_{lm} = (l+m)! O^{\star}_{lm}$$

the addition theorem (2.5) has to be modified for the elements of chargeless multipoles O_{lm} instead of the elements R_{lm}. This can be done by substituting the elements R_{lm} and yields

$$O_{lm}(\mathbf{a}+\mathbf{b}) = \sum_{j=0}^{l}\sum_{k=-j}^{j} O_{jk}(\mathbf{a})O_{l-j,m-k}(\mathbf{b})\,.$$

The addition theorem allows us to factorize the potential $1/|\mathbf{r}-(\mathbf{a}+\mathbf{b})|$ into a sum of triple products depending separately on $\mathbf{r}$, $\mathbf{a}$ and $\mathbf{b}$ as follows

$$\begin{aligned}\frac{1}{|\mathbf{r}-(\mathbf{a}+\mathbf{b})|} &= \sum_{l=0}^{\infty}\sum_{m=-l}^{l} O_{lm}(\mathbf{a}+\mathbf{b})M_{lm}(\mathbf{r}) \\ &= \sum_{l=0}^{\infty}\sum_{m=-l}^{l}\sum_{j=0}^{l}\sum_{k=-j}^{j} O_{jk}(\mathbf{a})M_{lm}(\mathbf{r})O_{l-j,m-k}(\mathbf{b})\,.\end{aligned}$$

The derivation for the multipole expansions ω_{lm} is straightforward

$$\omega_{lm}(\mathbf{a}+\mathbf{b}) = \sum_{j=0}^{l}\sum_{k=-j}^{j} \omega_{jk}(\mathbf{a})O_{l-j,m-k}(\mathbf{b})\,.$$

We identify operator $\mathcal{A}$ with

$$\mathcal{A}_{jk}^{lm}(\mathbf{b}) = O_{l-j,m-k}(\mathbf{b}). \tag{2.8}$$

This operator is also called Multipole2Multipole operator or M2M. Operator $\mathcal{A}$ is free of errors. Independent of the length of the multipole expansion, all shifted moments are exact since the operator only includes elements up to the order of the shifted ones. The M2M operator is a vertical operator acting on boxes of different tree levels.

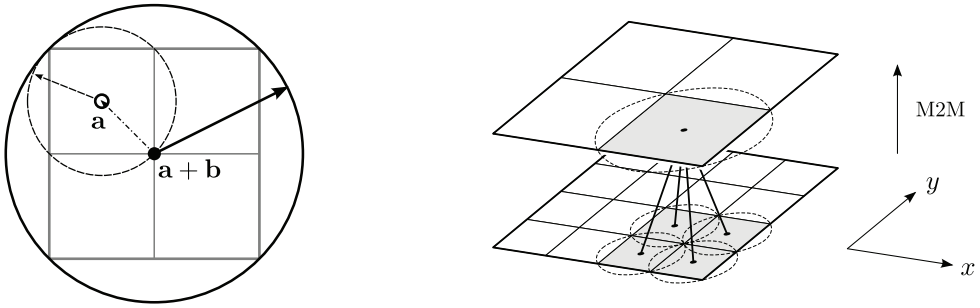
2.3.2. Conversion of a Multipole Expansion into a Local Expansion (M2L)

With help of the addition theorem in (2.6) we are able to transform an external multipole expansion into a local Taylor-like expansion. Since,

$$I_{lm}(\mathbf{r}) = \frac{1}{(l-m)!}M_{lm}(\mathbf{r})$$

we can substitute the irregular solid harmonics in (2.6) with the elements M_{lm} to derive a chargeless version of the operator

$$M_{lm}(\mathbf{b}-\mathbf{a}) = \sum_{j=0}^{\infty}\sum_{k=-j}^{j} M_{j+l,k+m}(\mathbf{b})O_{jk}(\mathbf{a})\,.$$



(a) Analytic Domain (b) Tree View

Figure 2.8.: M2M Operator for a two dimensional system. M2M is a vertical operator, exchanging information between levels.

Similarly to the first FMM operator we can substitute the terms O_{lm} and M_{lm} to obtain the charged version of this operator. Now, we are able to transform coefficients of an external multipole expansion around $\mathbf{a}$ to local Taylor-like coefficients around $(\mathbf{b}-\mathbf{a})$ as shown in figure 2.9. The transformed Taylor-like expansion can be represented as

$$\mu_{lm}(\mathbf{b}-\mathbf{a}) = \sum_{j=0}^{\infty}\sum_{k=-j}^{j} M_{j+l,k+m}(\mathbf{b})\omega_{jk}(\mathbf{a})$$

with Multipole2Local (M2L) operator

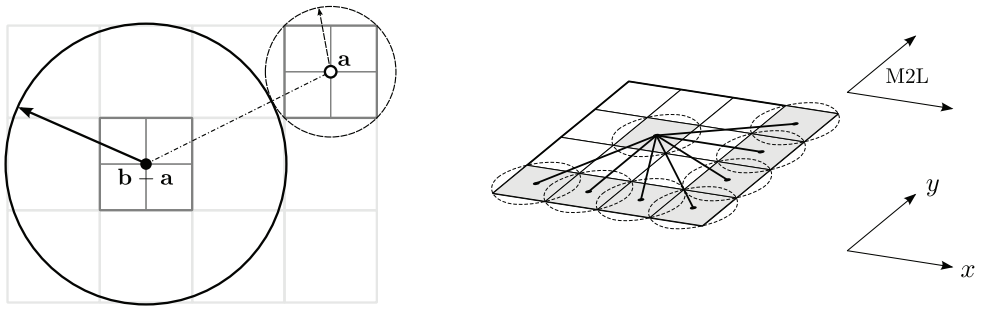
$$\mathcal{B}_{jk}^{lm}(\mathbf{b}) = M_{j+l,k+m}(\mathbf{b})\,. \tag{2.9}$$

Besides errors arising from the truncation of the expansion ω_{jk} with $j < p$, additional operator errors are introduced, because only $2p$ terms are considered in the sum for the elements $M_{j+l,k+m}$ with $j+l < 2p$. Since the operator itself is expanded up to $2p$ it allows to transform all available elements in ω_{jk}. The M2L operator is a horizontal operator acting on boxes of the same tree level.

2.3.3. Translation of a Local Expansion (L2L)

The last operator can be obtained by using (2.5) again. With the help of the factorized potential, given by

$$\begin{aligned}\frac{1}{|\mathbf{r}-(\mathbf{a}+\mathbf{b})|} &= \sum_{l=0}^{\infty}\sum_{m=-l}^{l} O_{lm}(\mathbf{a}+\mathbf{b})M_{lm}(\mathbf{r}) \\ &= \sum_{l=0}^{\infty}\sum_{m=-l}^{l}\sum_{j=0}^{l}\sum_{k=-j}^{j} O_{jk}(\mathbf{a})M_{lm}(\mathbf{r})O_{l-j,m-k}(\mathbf{b})\,,\end{aligned}$$



(a) Analytic Domain (b) Tree View

Figure 2.9.: M2L operator for a two dimensional system. M2L is a horizontal operator, exchanging information between boxes of the same level.

and the manipulation scheme for a double summation from (2.7) with elements

$$A_{jl}(\mathbf{a},\mathbf{b},\mathbf{r}) = \sum_{m=-l}^{l}\sum_{k=-j}^{j} O_{jk}(\mathbf{a})M_{lm}(\mathbf{r})O_{l-j,m-k}(\mathbf{b})$$

we can change the order of the summation

$$\frac{1}{|\mathbf{r}-(\mathbf{a}+\mathbf{b})|} = \sum_{l=0}^{\infty}\sum_{j=0}^{l} A_{jl}(\mathbf{a},\mathbf{b},\mathbf{r}) = \sum_{j=0}^{\infty}\sum_{l=j}^{\infty} A_{jl}(\mathbf{a},\mathbf{b},\mathbf{r})\,.$$

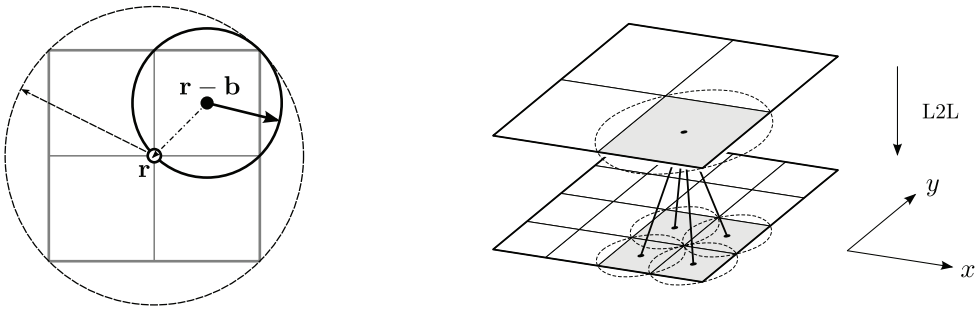
Relabeling the indices $l \leftrightarrow j, m \leftrightarrow k$ of the last equation and resubstituting the original multipole and Taylor-like expansion yields

$$\frac{1}{|\mathbf{r}-(\mathbf{a}+\mathbf{b})|} = \sum_{l=0}^{\infty}\sum_{j=l}^{\infty}\sum_{m=-l}^{l}\sum_{k=-j}^{j} O_{lm}(\mathbf{a})M_{jk}(\mathbf{r})O_{j-l,k-m}(\mathbf{b})\,,$$

Separating the terms $O_{j-l,k-m}$ and M_{jk} results in the operator $\mathcal{C}$, which allows us to translate a Taylor-like expansion located around $\mathbf{r}$ to its center at $(\mathbf{r}-\mathbf{b})$ as shown in figure 2.10 with

$$M_{lm}(\mathbf{r}-\mathbf{b}) = \sum_{j=l}^{p}\sum_{k=-j}^{j} O_{j-l,k-m}(\mathbf{b})M_{jk}(\mathbf{r})\,.$$

This operator is also called Local2Local (L2L) operator. Compared to operator $\mathcal{A}$, this operator introduces errors due to the finite representation of the multipole expansion. However, compared to operator $\mathcal{B}$ no additional operator errors arise. For a finite Taylor-like expansion with a



(a) Analytic Domain (b) Tree View

Figure 2.10.: L2L operator for a two dimensional system. L2L is a vertical operator, exchanging information between levels.

Operator	Purpose	Truncation Error	Additional Operator Error
$\mathcal{A}$	Multipole to Multipole	—	—
$\mathcal{B}$	Multipole to Local	•	•
$\mathcal{C}$	Local to Local	•	—

Table 2.1.: The three FMM operators have different sources of error. The most error-prone operator is operator $\mathcal{B}$. To derive valid error estimation schemes, the influence of operator $\mathcal{B}$ must be considered additionally to the truncation errors introduced by the finite expansion.

truncation at p poles the operator is exact with respect of the length of the Taylor-like expansion. The operator L2L is used to shift finite Taylor-like moments from a parent box to its children boxes' centers. The charged version of the operator is given by

$$\mu_{lm}(\mathbf{r}-\mathbf{b}) = \sum_{j=l}^{p} \sum_{k=-j}^{j} O_{j-l,k-m}(\mathbf{b}) \mu_{jk}(\mathbf{r})$$

with L2L operator

$$C_{jk}^{lm}(\mathbf{b}) = O_{j-l,k-m}(\mathbf{b})\,. \tag{2.10}$$

The L2L operator is a vertical operator acting on boxes of different tree levels.

Remark 2.3.1. Omitting operator M2M and L2L increases the complexity to $\mathcal{O}(N \log N)$. Such a scheme would behave like a Barnes-Hut treecode [12].

Obviously all operators induce a complexity of $\mathcal{O}(p^4)$. Especially high precision calculations are slowed down. To overcome this problem, improvements to the operators have been proposed.

2.3.4. Rotation-Based Operators

Unfortunately, the derived operators have a complexity of $\mathcal{O}(p^4)$. Especially for high precision calculation this is disadvantageous, because it will increase the amount of computation time. To circumvent the problem, White and Head-Gordon [31] proposed a different scheme improving the operator scaling to $\mathcal{O}(p^3)$. The improved scheme is easy to implement with the proposed standard FMM operators and only needs minor modifications. The increased memory usage is negligible and no additional approximations are induced. The original error bounds are retained. The rotation based operators are predicated on the observation that the three dimensional problem is reduced to a one dimensional problem if the translation or shift is carried out along the quantization axis (z-axis). Wigner rotation matrices are applied to rotate the multipole moments. A shift along the quantization axis with $\theta = 0$ and $\phi = 0$ yields a simplified form for the representation of the chargeless multipole moments with

$$O_{lm}(\mathbf{a}) = \frac{1}{(l+m)!} a^l \delta_{m0}$$

$$M_{lm}(\mathbf{r}) = (l-m)! \frac{1}{r^{l+1}} \delta_{m0}\,.$$

After substituting the new forms in the original operator formulae (2.8), (2.9), (2.10), we get

$$A_{jk}^{lm}(\mathbf{b}) = \frac{1}{(l-j+m-k)!} b^{l-j} \delta_{m-k,0}$$

$$B_{jk}^{lm}(\mathbf{b}) = (j+l-k-m)! \frac{1}{b^{j+l+1}} \delta_{k+m,0}$$

$$C_{jk}^{lm}(\mathbf{b}) = \frac{1}{(j-l+k-m)!} b^{j-l} \delta_{k-m,0}\,.$$

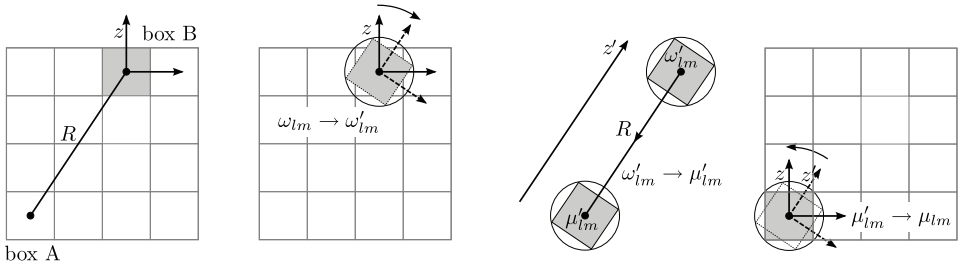
The new shifted and translated moments simplify to

$$O_{lm}(\mathbf{a}+\mathbf{b}) = \sum_{j=0}^{l} \frac{b^{l-j}}{(l-j)!} O_{jm}(\mathbf{a})$$

$$M_{lm}(\mathbf{a}-\mathbf{b}) = \sum_{j=0}^{\infty} \frac{(j+l)!}{b^{j+l+1}} O_{j,-m}(\mathbf{a})$$

$$M_{lm}(\mathbf{r}-\mathbf{b}) = \sum_{j=l}^{\infty} \frac{b^{j-l}}{(j-l)!} M_{jm}(\mathbf{r})\,.$$

Simple rotations preserve the total angular momentum. Therefore, any rotated spherical harmonic will be given as a linear combination of other spherical harmonics having the same order p. The implementation of the FMM presented here is based on these rotation-based operators.



(a) Interaction Set (b) Rotate ω_{lm} (c) Translate ω'_{lm} (d) Rotate back μ'_{lm}

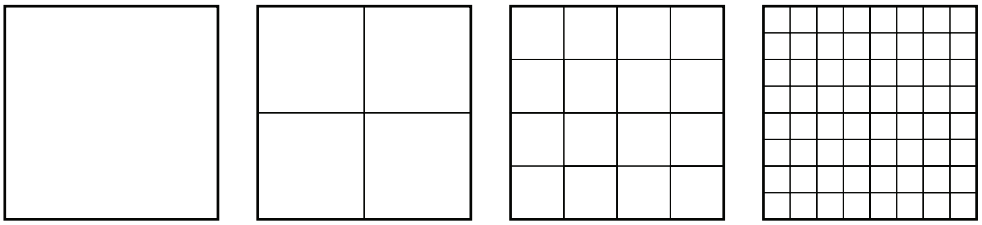
Figure 2.11.: (a) The box-box vector R between interacting box centers is usually not aligned along the z-axis. Therefore, the coordinate system is rotated (b) to align box B along the new z' axis. The multipole moments ω_{lm} transform into the rotated multipoles ω'_{lm} around the center of box B. (c) Operator $\mathcal{B}$ is used to translate the external multipole expansion ω'_{lm} into the local Taylor-like expansion μ'_{lm} around the center of box A. (d) Finally the Taylor-like expansion μ'_{lm} is rotated back into the original coordinate system yielding μ_{lm}.

2.3.5. Further Operator Compression

The presented rotation-based operators as well as the rotations themselves provide a complexity of $\mathcal{O}(p^3)$. However, efforts were made to further reduce the complexity of the operators, especially the translation operator $\mathcal{B}$. Compared to the vertical tree operators $\mathcal{A}$ and $\mathcal{C}$ which are invoked approximately once per box, the $\mathcal{B}$ operator is invoked 189 times for the smallest interaction set. A first attempt was made by Berman [32] with a grid-based approach, showing an operator complexity of $\mathcal{O}(p^2)$. Unfortunately, this method lacks analytic error bounds and demands an increased storage requirement. Another approach by Elliott and Board [33] facilitated a complexity of order $\mathcal{O}(p^2 \log p)$ via an FFT based approach. Unfortunately, this scheme needs additional memory due to the use of the Fast Fourier Transform. Other disadvantages are numerical instabilities and additional errors due to the use of the FFT. Yet another approach proposed by Cheng, Greengard and Rokhlin [34] is based on a combination of multipole expansions and exponential expansions (plane-wave expansions). Instead of expanding the potential $1/r$ into multipoles they use the following form

$$\frac{1}{r} = \frac{1}{2\pi} \int_0^\infty e^{-\lambda z} \int_0^{2\pi} e^{i\lambda(x\cos\alpha + y\sin\alpha)} \mathrm{d}\alpha \mathrm{d}\lambda \,.$$

The integrals are solved numerically for certain accuracies. The translation from a multipole in exponential form to a local expansion in exponential form comprises $\mathcal{O}(p^2)$ operations. To produce an exponential expansion in the first place operations with costs $\mathcal{O}(p^3)$ are required. However, the exponential expansion depends on the direction of the interaction. Therefore, six



(a) Level 1 – Depth 0 (b) Level 2 – Depth 1 (c) Level 3 – Depth 2 (d) Level 4 – Depth 3

Figure 2.12.: The FMM tree is subdivided until a certain number of particles on the lowest level is reached. Level $L = 1$ corresponds to the highest level, level L_{max} corresponds to the lowest level. The subdivision shown, adds four times the number of boxes to the tree for each level. In three dimensions, the number of boxes increases by a factor of 8.

different operators in 3D are needed. Since this scheme also involves additional errors from the approximation of the exponential expansion it was not considered for our implementation.

2.4. $\mathcal{O}(N \log N)$ Algorithm

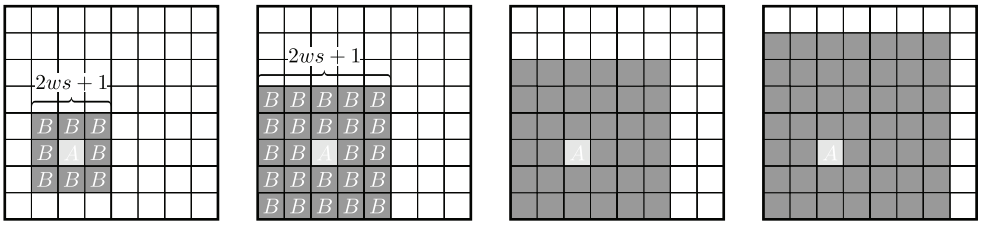
In this section we describe the details of the simpler algorithm not using any translation operators which yields an overall complexity of $\mathcal{O}(N \log N)$. However, the final FMM scheme can be based on this approach. We start to enclose our given particles inside a cube and call this box *simulation box*. Next, we introduce a hierarchy of boxes by subdividing this simulation box by half along each axis. The refinement level of the entire simulation box is named level one ($L = 1$). An arbitrary level of refinement $L + 1$ can be obtained by subdividing level L into eight equal subboxes. The eight subboxes at level $L + 1$ are called child boxes of their parent box at refinement level L. The scheme corresponds to the construction of an oct-tree depicted in figure 2.12.

Now we can derive subsets of boxes via the following definitions.

Definition 2.4.1. *Two boxes A and B are called* next neighbors *if they are at the same tree level and box B is enclosed by a box of size* $(2ws + 1)^3$ *around the center of A with a box edge length of one. Next neighbors interact in the near field (NF).*

The former definition allows us to increase the range for the near field part to the full simulation box ($ws \to \infty$) yielding a direct interaction scheme with $\mathcal{O}(N^2)$ complexity. A definition for a minimal $ws = 1$ criterion can written as follows:

Definition 2.4.2. *Two boxes are called* next neighbors *if they are at the same tree level and share a boundary point (see figure 2.14b). Next neighbors interact in the near field (NF).*



(a) $ws = 1$ (b) $ws = 2$ (c) $ws = 3$ (d) $ws = 4$

Figure 2.13.: FMM well separateness criterion. For increasing ws the FMM leads towards the direct summation scheme with unfavorable $\mathcal{O}(N^2)$ complexity. All nearest neighbors B for box A are shown. In figure (c) and (d) the nearest neighbors to the left and the bottom of A already exceed the boundary of the simulation box.

The different interaction sets for the near field can be seen in figure 2.13. After defining the near field, the far field has to be defined accordingly.

Definition 2.4.3. *Two boxes A and B are called* well separated *if they are at the same tree level and are not next neighbors (see figure 2.14c). Well separated boxes interact in the far field (FF).*

The last definition does not limit the number of interactions in the far field, therefore we have to set up a confined interaction list.

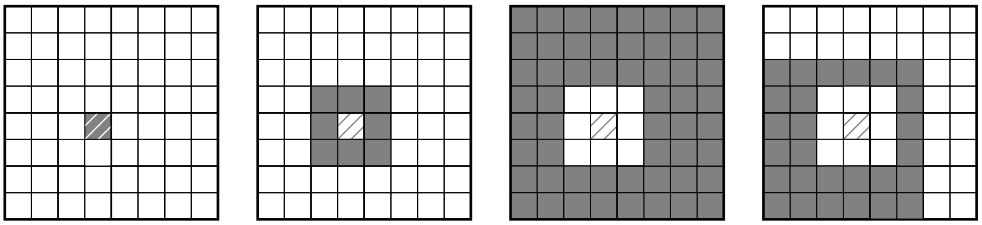
Definition 2.4.4. *An* interaction list i *is associated with each box A, consisting only of children of the next ws neighbors of A's parent which are well separated from box A (see figure 2.14d). The interaction list limits the number of far field (FF) interaction for each box.*

These definitions allow us to compute interactions between a constant number of boxes on each level.

The spatial refinement enables us to cluster together particles from one and the same box into multipole moments. The expansion of the particles is performed around the box center. With the help of definition 2.4.4 it is possible to define interaction sets for multipole expansions of these boxes.

Definitions 2.4.3 and figure 2.12 show clearly that there are no interactions on level 1 and level 2 since these levels do not contain two separated boxes, hence all boxes are nearest neighbors.

Starting on level 3 with its 64 boxes we can use multipole expansions to compute interactions between particles of a source box with a multipole expansion of particles in a remote box. The error bound connected with these interactions will be discussed later in section 2.7. After calculating the interactions on level 3, we can use a recursion scheme to include boxes on the next refinement level (level 4). After subdividing all boxes on level 3, we again identify the



(a) Box – Near field (b) Nearest Neighbors interaction set (c) Full far field interaction (d) Far field interaction set

Figure 2.14.: Different interaction sets for fixed separation criterion $ws = 1$. (a) All particles within one box (on the lowest level) interact directly. (b) Next neighbor boxes (on the lowest level) interact directly, since the convergence of the expansions can not be guaranteed or may converge slowly. (c) Boxes within the interaction list (on each level) interact via multipoles.

interaction set for each box. Since we already accounted for all interactions of boxes outside the parent boxes' next neighbors we must ignore these interactions. The scheme is repeated for every tree level until we reach the lowest level. On the lowest level we calculate the missing interactions for the direct neighboring boxes via a direct calculation. In the original work of Rokhlin and Greengard the recursive process of refining is halted roughly after $\log_8 N$ levels assuming a homogeneous particle distribution.

Computational Complexity

The complexity of the $\mathcal{O}(N \log N)$ scheme can be derived as follows. Each particle contributes p^2 coefficients to each box multipole expansion independent of the box size. Therefore, we need $p^2 N$ operations to set up box multipole expansions on each level. The maximal interaction list of each box consists of 189 boxes ($ws = 1$). If we assume to set up the multipole expansions for every single particle separately (on the lowest level), $189Np^2$ operations are needed. Assuming a homogeneous particle distribution on $\log_8 N$ levels, we end up with $8^{\log_8 N} = N$ boxes on the lowest level. The homogeneous distribution results in $\mathcal{O}(1)$ particles per box. Since not all particles interact via multipoles we have to compute the near field parts directly requiring $27N$ operations. Therefore, the total costs for this scheme sum up to

$$189Np^2 \log_8 N \cdot \mathcal{O}(1) + 27N \cdot \mathcal{O}(1)\,.$$

This scheme can be referred as arbitrary precision Barnes-Hut [12] scheme with error bounds. The asymptotic complexity has improved from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$. Unfortunately, the speedup compared to a direct summation is only modest, since we have to set up multipole moments on every level separately and do not use information from higher or lower levels.

2.5. $\mathcal{O}(N)$ Algorithm

With the help of the introduced operators M2M, M2L and L2L it is possible to derive a scheme with overall complexity $\mathcal{O}(N)$. A multipole expansion of a source box does not have to be evaluated for any particle in the target box. Instead we transform all far field multipole expansions into a single local expansion in the target box. Then, the local expansion is evaluated at each individual target particle position.

General Workflow

The original FMM proposed by Greengard and Rokhlin can be sectioned into five main steps. The determination of the FMM parameter set, is influenced only by a worst-case error scheme, the actual positions of the particles are not taken into account. We briefly want to outline the single steps:

- Preprocessing steps
  - Define the separation criterion ws
  - Define the order of multipoles p for a given precision ΔE
  - Define the depth of the FMM tree d
  - Expand particles into multipole moments ω_{lm} on the lowest level
- Pass 1 ▷ Translate multipole moments ω_{lm} up the tree
- Pass 2 ▷ Transform multipole moments ω_{lm} into Taylor moments μ_{lm}
- Pass 3 ▷ Translate Taylor moments μ_{lm} down the tree
- Pass 4 ▷ Compute the far field contributions of the system
  - Compute the far field potential $\Phi_{\text{FF}}(x, y, z)$
  - Compute the far field forces $\mathbf{F}_{\text{FF}}(x, y, z)$
  - Compute the far field energy E_{FF}
- Pass 5 ▷ Compute the near field contributions of the system
  - Compute the near field potential $\Phi_{\text{NF}}(x, y, z)$
  - Compute the near field forces $\mathbf{F}_{\text{NF}}(x, y, z)$
  - Compute the near field energy E_{NF}

The outlined workflow is now described in detail in the following sections.

Preliminary Steps

The coordinates are scaled into a $[0, 1] \times [0, 1] \times [0, 1]$ unit simulation box to guarantee numerical stability of the computations and to simplify the involved FMM operators. Since the FMM allows *a priori* error bounds the order of poles p can be set-up for a given separation criterion. Let us assume for the moment a separation criterion $ws = 1$. A higher value of ws would yield a faster convergence of the expansion (less multipole terms are needed) but higher costs in the near field computations. Since this part scales quadratically in N, we want to eliminate as many

(a) Particles inside a box A

(b) Particles inside A's neighbor boxes

Figure 2.15.: In general it is not possible to set ws to zero, since neighboring boxes may have particles overlapping with the multipole expansion of a neighboring box. However, the convergence of the expansion demands separation. Therefore, next neighbors must not interact via a multipole expansion.

direct interactions as possible, therefore ws has to be set to one; it's minimal value. The reason for this minimum ws is shown in figure 2.15.

After defining the order of poles p, we are able to set up the FMM tree. The refinement of the tree stops, once a certain number of particles in the lowest level is reached (e.g. k particles per box, $k \ll N$). The particles are sorted via a radix sort [35] into the lowest level boxes. Now we expand all particles in each lowest level box into multipole moments about the center of the same box.

Pass 1

Assuming we have more than 3 levels in the oct-tree, we now shift the multipole coefficients to the center of the parent box via the M2M operator. Since each parent box consists of 8 child boxes (in 3D) the moments of the expansion can be summed up at the new center and are stored as moments of the parent box. The scheme is repeated until level 3 is reached. Now we have a multipole expansion for each box on every level (starting at level 3).

Each child expansion at the center of the box at $\mathbf{a}_i$ is shifted to the center of the common parent box at $\mathbf{a} + \mathbf{b}$ and then summed up with the shifted expansion of the other seven child

boxes.

$$\begin{aligned}
\omega^1_{lm}(\mathbf{a}_1 + \mathbf{b}_1) &= \sum_{j=0}^{l} \sum_{k=-j}^{j} \mathcal{A}^{lm}_{jk}(\mathbf{b}_1)\omega^1_{jk}(\mathbf{a}_1) \\
&\vdots \\
\omega^8_{lm}(\mathbf{a}_8 + \mathbf{b}_8) &= \sum_{j=0}^{l} \sum_{k=-j}^{j} \mathcal{A}^{lm}_{jk}(\mathbf{b}_8)\omega^8_{jk}(\mathbf{a}_8) \\
\omega_{lm}(\mathbf{a} + \mathbf{b}) &= \sum_{i=1}^{8} \omega^i_{lm}(\mathbf{a}_i + \mathbf{b}_i)
\end{aligned}$$

Pass 2

In the second pass we apply a modification of the already known scheme from the described $\mathcal{O}(N \log N)$ scheme. Instead of bringing every particle in a given target box to interaction with all multipoles in the interaction set, we transform (at most) 189 source multipole moments within each interaction set into local Taylor-like moments for each target box on each level.

$$\begin{aligned}
\mu^1_{lm}(\mathbf{b}_1 - \mathbf{a}_1) &= \sum_{j=0}^{p} \sum_{k=-j}^{j} \mathcal{B}^{lm}_{jk}(\mathbf{b}_1)\omega^1_{jk}(\mathbf{a}_1) \\
&\vdots \\
\mu^i_{lm}(\mathbf{b}_i - \mathbf{a}_i) &= \sum_{j=0}^{p} \sum_{k=-j}^{j} \mathcal{B}^{lm}_{jk}(\mathbf{b}_i)\omega^i_{jk}(\mathbf{a}_i) \\
\mu_{lm}(\mathbf{b} - \mathbf{a}) &= \sum_{i=1}^{\text{ilist}} \mu^i_{lm}(\mathbf{b}_i - \mathbf{a}_i)
\end{aligned}$$

Pass 3

The third pass shifts the Taylor-like moments starting from level 3 to the lowest level via the L2L operator. On the way downwards, the Taylor-like moments of the actual level are summed up with the shifted interactions from a higher level. The following equations show a shift of a

Taylor-like expansion μ_{jk} at level L to expansions μ_{lm}^{i} at level $L+1$

$$\mu_{lm}^{1}(\mathbf{r}-\mathbf{b}_1) = \sum_{j=0}^{p}\sum_{k=-j}^{j} \mathcal{C}_{jk}^{lm}(\mathbf{b}_1)\mu_{jk}(\mathbf{r})$$

$$\vdots$$

$$\mu_{lm}^{8}(\mathbf{r}-\mathbf{b}_8) = \sum_{j=0}^{p}\sum_{k=-j}^{j} \mathcal{C}_{jk}^{lm}(\mathbf{b}_8)\mu_{jk}(\mathbf{r})\,.$$

Pass 4

The fourth pass finally computes the interactions between the Taylor-like moments representing all effects of all well separated particles and the particles at positions $\mathbf{a}_i = (a_i, \alpha_i, \beta_i)$ inside the actual target box

$$\Phi_{\text{FF}}(\mathbf{a}_i) = \sum_{l=0}^{p}\sum_{m=-l}^{l} \mu_{lm}(\mathbf{r}) a_i^l \tilde{P}_{lm}(\cos\alpha_i) e^{-im\beta_i}$$

$$\mathbf{F}_{\text{FF}}(\mathbf{a}_i) = -\sum_{l=0}^{p}\sum_{m=-l}^{l} \mu_{lm}(\mathbf{r}) \nabla_{\mathbf{a}_i}\left[a_i^l \tilde{P}_{lm}(\cos\alpha_i) e^{-im\beta_i}\right]$$

$$E_{\text{FF}} = \sum_{l=0}^{p}\sum_{m=-l}^{l} \mu_{lm}(\mathbf{r})\omega_{lm}(\mathbf{a})\,.$$

Pass 5

The fifth pass calculates the neglected interactions, hence all interactions which cannot be evaluated via far field expansions because of the chosen separation criterion ws. Since we ensured that the number of particles per box M on the lowest level is independent of the total number of particles N the costs of this pass is $\mathcal{O}(MN)$ with $M \ll N$. Pass 5 contains all interactions from particles M_{ibox} within one box *ibox*

$$\Phi(\mathbf{r}_i) = \sum_{j=1}^{M_{\text{ibox}}} \frac{q_j}{r_{ij}} \quad (i \neq j)$$

$$\mathbf{F}(\mathbf{r}_i) = q_i \sum_{j=1}^{M_{\text{ibox}}} \frac{q_j}{r_{ij}^3}\mathbf{r}_{ij} \quad (i \neq j)$$

$$E_{\text{NF}}^{\text{ibox}} = \sum_{i=1}^{M_{\text{ibox}}-1}\sum_{j=i+1}^{M_{\text{ibox}}} \frac{q_i q_j}{r_{ij}} \quad (i \neq j)$$

and particles M_{jbox} from all nb neighboring boxes *jbox* on the lowest tree level

$$\Phi(\mathbf{r}_i) = \sum_{k=1}^{\text{nb}} \Phi_k(\mathbf{r}_i) = \sum_{k=1}^{\text{nb}} \sum_{j=1}^{M_{\text{jbox(k)}}} \frac{q_{jk}}{r_{ijk}}$$

$$\mathbf{F}(\mathbf{r}_i) = q_i \sum_{k=1}^{\text{nb}} \mathbf{F}_k(\mathbf{r}_i) = q_i \sum_{k=1}^{\text{nb}} \sum_{j=1}^{M_{\text{jbox(k)}}} \frac{q_{jk}}{r_{ijk}^3} \mathbf{r}_{ijk}$$

$$E_{\text{NF}}^{\text{jbox}} = \sum_{i=1}^{M_{\text{ibox}}} \sum_{k=1}^{\text{nb}} \sum_{j=1}^{M_{\text{jbox(k)}}} \frac{q_i q_{jk}}{r_{ijk}} .$$

Computational Complexity

Let us derive the computational complexity of the presented algorithm. To set up the multipole moments in the first pass we need $\mathcal{O}(p^2)$ operations for each particle, which sums up to $N \cdot \mathcal{O}(p^2)$ for the entire system. To shift the multipole moments up the tree, a rough estimate for the number of tree level is necessary. Clearly the number of operations increases with increasing number of levels L, therefore we assume a lowest level with only one particle per box. The work on the lowest level is proportional to 8^{L-1}. On the next higher level the work is reduced by a factor of eight yielding 8^{L-2}. Since we assumed one particle per box on the lowest level, the total number of boxes is given by

$$\sum_{i=0}^{L-1} 8^i = \frac{8^L - 1}{7} = \frac{8N - 1}{7} .$$

The work per particle hereby does not depend on the number of particles N but is constant

$$\lim_{N \to \infty} \frac{8N - 1}{7N} = \frac{8}{7} .$$

Therefore, the complexity of Pass 1 depends linearly on the number of particles N and needs about $\frac{8}{7}N \cdot \mathcal{O}(p^4)$ operations. Using the rotation-based approach reduces the complexity to $\frac{8}{7}N \cdot \mathcal{O}(p^3)$.

The translation of the multipole moments to Taylor-like moments in Pass 2 is done on every level but only for boxes within the interaction set. The largest sets contains 189 boxes ($ws = 1$). Therefore, the complexity is $\frac{8}{7}189N \cdot \mathcal{O}(p^4)$ or $\frac{8}{7}189N \cdot \mathcal{O}(p^3)$ applying the rotation based approach.

The third pass has the same complexity as the first pass, since the Taylor-like moments are shifted along the tree levels in the opposite direction. Therefore it results in a complexity of $\frac{8}{7}N \cdot \mathcal{O}(p^4)$ or $\frac{8}{7}N \cdot \mathcal{O}(p^3)$ depending on the complexity of the M2L operator.

Pass 4 requires a work proportional to Np^2, since for every particle we have to calculate the interaction from the Taylor-like expansion with $\mathcal{O}(p^2)$ coefficients.

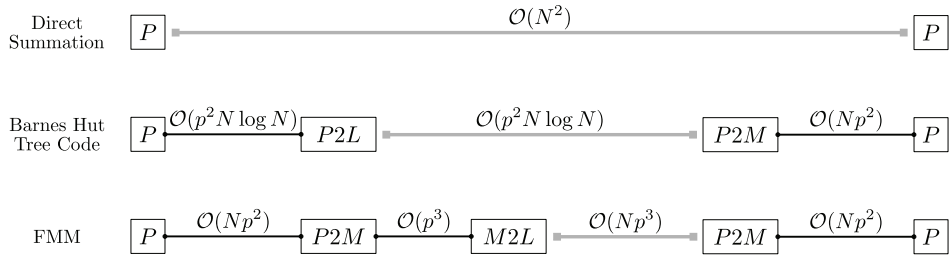


Figure 2.16.: Comparison of the complexity of three different methods. Direct Interaction, Barnes-Hut Treecodes and the FMM. The black lines represent steps in the algorithm where particles or particle collections are transformed into expansions. The gray lines represent the actual interaction.

Pass 5 has a quadratical complexity, however, the number of neighbor boxes and therefore the number of particles in the near field is limited (one per box). Hence, the costs of this pass are $27N$. The overall complexity for the original approach calculates to

$$\frac{8}{7}191Np^4 \cdot \mathcal{O}(1) + 2Np^2 \cdot \mathcal{O}(1) + 27N \cdot \mathcal{O}(1)$$

and for the rotation-based approach calculates to

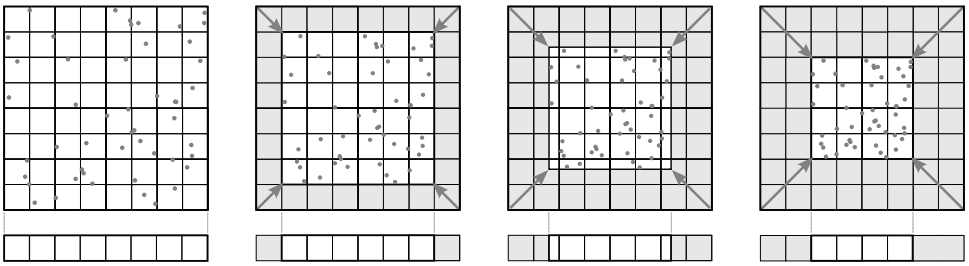
$$\frac{8}{7}191Np^3 \cdot \mathcal{O}(1) + 2Np^2 \cdot \mathcal{O}(1) + 27N \cdot \mathcal{O}(1)\,.$$

A comparison of the asymptotical complexity of the FMM and the related Barnes-Hut algorithm as well as the direct summation is depicted in figure 2.16.

2.6. Implementation Details

2.6.1. Fractional Tree Depth

For homogeneously distributed particle systems the FMM algorithm has to add a new level to the tree, if the number of particles is increased by a factor of 8 to retain the linear scaling. For fewer additional particles the tree is not changed resulting in a locally quadratical behavior as shown in figure 2.18b. This effect can be reduced by introducing fractional levels [36]. Therewith it is possible to generate an arbitrary number of lowest level boxes which allows a better balance between the near and far field computations. Therefore, the simulation box is scaled (screwed towards the center) to generate the desired number of boxes. This scaling is shown in figure 2.17. For flat trees with few levels the number of fractional levels is limited. For deep trees however, the number of possible fractional levels increases and a useful balance can be obtained



(a) $d_f = 3.000$ (b) $d_f = 2.585$ (c) $d_f = 2.263$ (d) $d_f = 2.000$

Figure 2.17.: (a) The fractional tree depth d_f is identical to the tree depth $d = 3.000$. (b) The fractional tree depth $d_f = 2.585$ is smaller than the tree depth $d = 3.000$. The fractional tree depth in (c) does not coincide with a subset of boxes. Therefore, it creates the same amount of empty boxes as (b). However, the workload may be different since the particles occupy different boxes. (d) The fractional tree depth $d_f = 2.000$ represents a tree depth of $d = 2.000$. The gray boxes do not hold particles and therefore can be pruned from the calculation.

more easily as shown in figure 2.18a. To calculate the fractional depth d_f or fractional level L_f, we use

$$d_f = \log_2 N_b \quad \text{and} \quad L_f = d_f + 1 = \log_2 N_b + 1$$

with N_b being the number of non-empty (fractional) boxes in one dimension. The fractional depth d_f can be obtained by minimizing the workload of both the near field and far field and is described in section 2.6.3. It strongly depends on the user-requested error bound, the total number of particles and the particle distribution of the system.

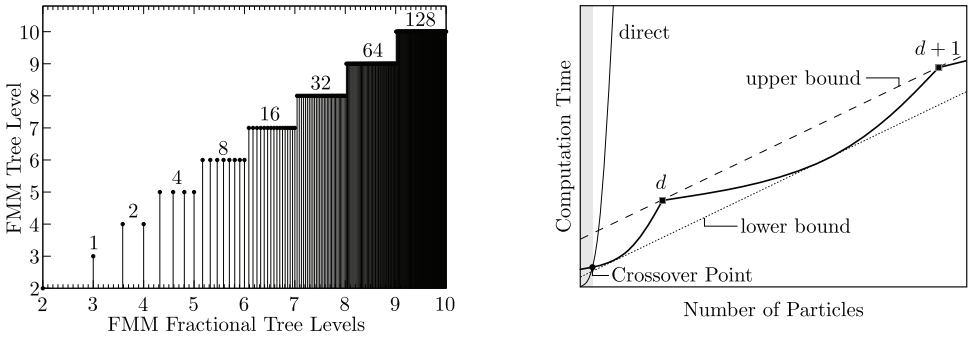
2.6.2. Data Structures

FMM Oct-tree

To store and manage the different boxes on different levels we use an oct-tree [37] as data structure. It allows to hierarchically decompose three-dimensional data. The root node of the tree contains all particles in the simulation box. Every recursively derived internal node (child box) contains one octant of the parent node (parent box). The oct-tree is an extension to binary trees in one dimension and quadtrees [38] in two dimensions. Thus, the spatial decomposition of the simulation box is realized in a uniform manner.

Pruning Empty Boxes

To store a full FMM tree would be disadvantageous for nonuniform particle distributions. Not only a huge amount of memory (8^L boxes per level L) is needed to store the full tree `FMMtree`,



(a) Available fractional levels L_f for every level

(b) $\mathcal{O}(N)$ scaling with locally quadratical behavior

Figure 2.18.: The left figure (a) shows the density distribution of available intermediate fractional levels for levels 2 to 10. The number of available fractional levels doubles with each additional tree level. The right figure (b) shows the upper and lower bound of the scaling without fractional levels. Applying the fractional level scheme yields a scaling curve near the lower bound. The gray area represents the range of particles in which the direct summation is faster than the FMM. The black dot represents the crossover point for the FMM with the direct summation for a certain precision ε. The black squares represent a certain number of particles for which an additional tree level is introduced.

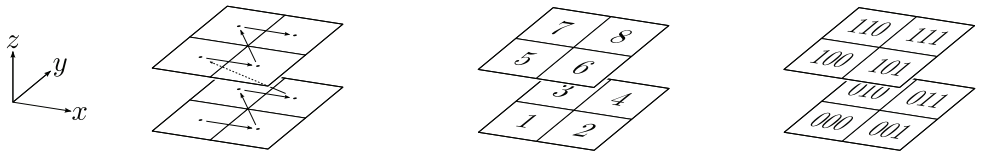
but also a fragmented memory layout due to empty boxes will arise. The memory layout for such an implementation looks like:

```
integer :: treedim = 2**Lmax
integer, dimension(treedim,treedim,treedim) :: FMMtree

boxnr = FMMtree(x,y,z)
```

In order to provide a memory efficient FMM scheme, boxes containing no particles are pruned from the data structures. This also ensures small memory fragmentation. Additionally the memory consumption of all data structures not directly depending on the number of particles N should be limited in order to have a memory complexity of order N:

```
integer :: N
integer, dimension(N) :: FMMsmalltree

boxnb = FMMsmalltree(addr)
```

The box vector `FMMsmalltree` only holds non-empty boxes. To obtain the box number of a certain particle, the list `FMMsmalltree` is accessed with the specific particle number. To generate this one-dimensional list for all particles we use a space-filling curve scheme to reduce the three dimensional space of box coordinates to a one-dimensional.



(a) Z-Ordering (b) Integer box numbers (c) Binary box numbers

Figure 2.19.: (a) Ordering of boxes in the FMM tree according to a Morton (Z)-curve. Figure (b) shows the integer representation of the box numbers resulting from a Z-ordering. Figure (c) shows the binary representation of the box numbers.

Space-Filling Indexing

A space-filling curve (SFC) [39] allows us to continuously connect all sub-boxes on a certain level of the FMM tree. A lot of different curves are proposed to fill space. The most commonly used curves are the Hilbert-Peano-, Gray- and Morton-SFC. Depending on the required properties, some curves have advantages (locality, jumps) compared to others. A discussion of important properties can be found in [40]. Since we have two FMM operators moving multipole information between levels, it would be advisable to have a SFC scheme allowing to quickly generate parent or child box numbers. Therefore, a Morton-ordered, i.e. Z-ordered SFC was chosen to map the 3D box structure depicted in figure 2.19. The access pattern for the tree therefore has to change. First, we have to calculate a SFC index `xyz` from the coordinates via a separate function `calxyz`. Afterwards, the SFC address can be used to access the tree-structure `FMMsmalltree`:

```
integer :: N
integer, dimension(N) :: FMMsmalltree

xyz = calxyz(x,y,z)
if(xyz /= nonempty) then
  boxnb = FMMsmalltree(xyz)
end if
```

Accessing Tree Nodes

The advantage of the Morton-ordered SFC scheme for FMM-based algorithms comes from a straightforward box generation scheme. Let us assume we want to obtain the box numbers of a parent box with binary representation 110101010. The 8 children can be generated by bit-shifting the binary representation of the parent box by three digits to the left. The last three digits are used to name the child boxes starting from 000 up to 111. Therefore, the box numbers of the childboxes yield 110101010000 up to 110101010111. Starting from the root-node (simulation box) containing all particles the FMM tree is set up. Therefore, the full FMM tree

	Dimension			Mixing	SFC Index
	x	y	z		
(x,y,z)	$x_3\,x_2\,x_1$	$y_3\,y_2\,y_1$	$z_3\,z_2\,z_1$	$x_3\,y_3\,z_3\,x_2\,y_2\,z_2\,x_1\,y_1\,z_1$	xyz
(0,2,2)	**0 0 0**	0 1 0	*0 1 0*	**0** 0 *0* **0** 1 *1* **0** 0 *0*	24
(3,0,7)	**0 1 1**	0 0 0	*1 1 1*	**0** 0 *1* **1** 0 *1* **1** 0 *1*	111
(7,2,0)	**1 1 1**	0 1 0	*0 0 0*	**1** 0 *0* **1** 1 *0* **1** 0 *0*	308

Table 2.2.: Morton space-filling curve and bit mixing in three dimensions. Each box is represented by its coordinates (x, y, z). This three dimensional representation can be transformed to a Morton ordered one-dimensional box index (SFC index).

is stored implicitly by only storing the lowest level L_{max}. All parent boxes can be generated by performing a right-shift and thereby dropping the last 3 digits in the binary representation of the box number. The function `calxyz` has to perform an interweave operation to generate the SFC box number. This scheme is depicted in table 2.2.

Multipole/Taylor-Coefficients Symmetry

A large amount of memory is used by the multipole and Taylor-like expansions. The coefficients are stored separately in double precision variables for real and imaginary part, each part consuming 8 bytes of memory (one "memory element"). These moments have to be stored for every non-empty box up to order p, they consume $(p+1)^2$ "memory elements" per box (see figure 2.20a). Since the associated Legendre polynomials for negative m are defined by

$$P_{l,-m}(x) = (-1)^m \frac{(l-m)!}{(l+m)!} P_{lm}(x)\,.$$

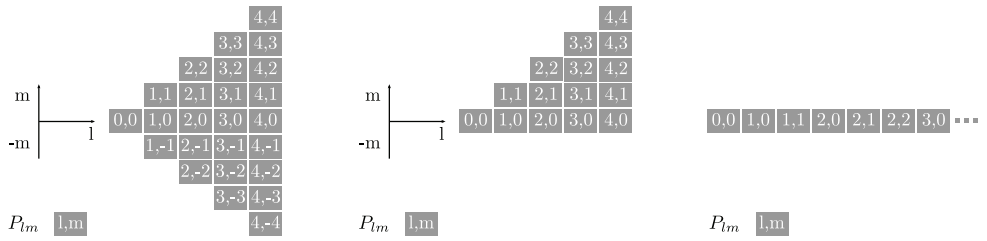
We can use this definition to generate the scaled associated Legendre polynomials $\tilde{P}_{lm}$ and $\tilde{\tilde{P}}_{lm}$ for negative m with

$$\begin{aligned}\tilde{P}_{l,-m} &= (-1)^m \tilde{P}_{lm} \\ \tilde{\tilde{P}}_{l,-m} &= (-1)^m \tilde{\tilde{P}}_{lm}\,.\end{aligned}$$

Therefore, the redundancy in the multipole and Taylor-like moments can be reduced using

$$\begin{aligned}O_{l,-m} &= (-1)^m O^*_{lm} \\ M_{l,-m} &= (-1)^m M^*_{lm}\,.\end{aligned}$$

The reduced set of coefficients only uses $\frac{1}{2}(p+1)(p+2)$ "memory elements" per non-empty box, which can be seen in figure 2.20 and table 2.3.



(a) Full Set (b) Reduced Set (c) Memory Order

Figure 2.20.: The full set of multipole moments (a) can be stored more efficiently without redundant information (b). To allow linear memory access the multipoles are stored as a 1D-list (c) not an 3D-array.

Memory Footprint

It is important for an $\mathcal{O}(N)$ algorithm to limit the use of memory to $\mathcal{O}(N)$. Therefore, all data types, especially those containing tree information data need to be limited. However, the access of neighboring boxes and interaction sets have to be performed frequently. As a compromise the full tree (regardless of empty boxes) is stored up to a certain level L_m. For trees with more levels $L_{\max} > L_m$ a different scheme is used. On today's general purpose computers we found a reasonable limitation for the full tree at 128MB or 1GB. Especially inhomogeneous particle distributions benefit from this modified memory access scheme. An overview of the most memory consuming data structures are given in table 2.3.

2.6.3. Minimization of Near Field and Far Field Computation Time

Embedded in the error control scheme the optimal tree depth L is determined prior to the FMM calculation. For a certain order of poles p and a certain level L, the workload for the far field and near field can be computed. Since the workload for the near field computation decreases with each additional level (see figure 2.21a) and the workload for the far field increases with each additional level (see figure 2.21b), a crossover point will establish an optimal tree level L_{opt} with minimal computation time (see figure 2.21c). The following scenario for the near field may occur with an increasing number of levels. All associated labels in the list can be found in figure 2.21a.

1. The workload decreases from level L to level $L + 1$.
2. The workload remains constant from level L to level $L + 1$.
3. The workload becomes zero from level L to level $L + 1$ and stays zero for all additional levels.

Type	Scope	Data Type	Number of Elements	Element Size
input	coordinates	`double`	$3N$	8 bytes
input	charges/masses	`double`	N	8 bytes
output	Cartesian forces	`double`	$3N$	8 bytes
output	Coulomb potential	`double`	N	8 bytes
aux.	box vector	`integer`	N	8 bytes
aux.	box scratch vector	`integer`	N	8 bytes
aux.	box sort vector	`integer`	N	8 bytes
aux.	FMM tree level*	`integer`	8^L	8 bytes
aux.	multipole expansion	`double`	$\frac{1}{2}(p+1)(p+2)b$	16 bytes
aux.	Taylor expansion	`double`	$\frac{1}{2}(p+1)(p+2)b$	16 bytes

* The full tree level is only stored for trees up to level $l = 8$. Afterwards a different access scheme is used to avoid non-linear scaling of the memory.

Table 2.3.: Memory requirements for input, output and auxiliary data necessary for the FMM. The variable b represents the number of all non-empty boxes on all tree levels and therefore depends on the particle distribution. The expansions are stored in two `real` arrays instead of an single array of type `complex`.

For the far field contributions a similar observation can be determined. All associated labels in the list can be found in figure 2.21b.

1. The workload is zero from level L to level $L+1$ depending on the ws.
2. The workload increases from level L to level $L+1$.
3. The workload remains constant after a certain level L.

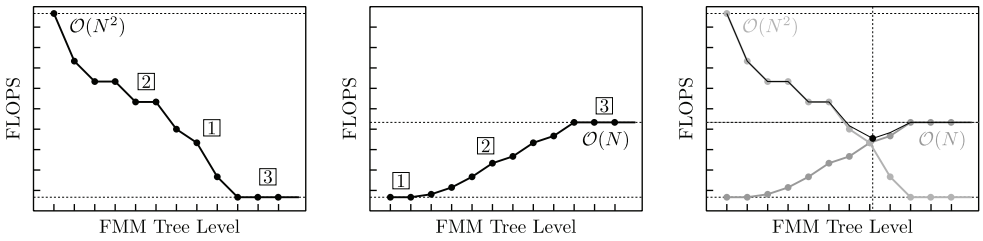
2.6.4. Limitations of the Algorithm

In contrast to the classical direct summation scheme the runtime of the FMM depends on the positions of the particles. We identify the limitations with respect to the depth of the tree and the order of poles.

FMM Tree Depth Dependencies

The linear complexity is only possible for systems with homogeneously distributed particles. The workload W has to be distributed according to a geometric series

$$W = \sum_{n=0}^{\infty} \frac{1}{a^n} \quad (a > 1,\ n \in \mathbb{N}_0)$$



(a) Near field (b) Far field (c) Both Contributions

Figure 2.21.: The minimal workload (optimal FMM treedepth L) depends on the computational load in the (a) near field and (b) far field for a certain order of poles p. The total workload is represented by the black line in figure (c).

in order to be $\mathcal{O}(N)$. For a homogeneously distributed particle system the workload on the lowest level L_{max} is set to one. The next level $L_{\text{max}} - 1$ has a workload of $1/8$, the next level $L_{\text{max}} - 2$ has a workload of $1/64$. The workload can be represented as a geometric series with $a = 8$ for an ideal homogeneously distributed particle system in three dimensions. For purely homogeneous 2D problems the workload W scales accordingly with $a = 4$, purely homogeneous 1D systems will generate a workload W corresponding to $a = 2$.

For heavily clustered systems the complexity may increase to $\mathcal{O}(N^2)$. An example can be constructed as shown in figure 2.22, in which the tree has to add a new level with each new cluster of size k. Depending on the implementation k can also be reduced to 1. The workload, i.e. number of occupied boxes does not decrease by a constant factor, but by a term $k \cdot i$. Therefore, the tree depth depends linearly on the number of particles, creating a $\mathcal{O}(N^2)$ complexity

$$W = k \sum_{i=0}^{N/k} i = \frac{1}{2} k \frac{N}{k} \left(\frac{N}{k} + 1 \right) \sim \mathcal{O}(N^2) \,.$$

For arbitrary particle systems there is no asymptotical boundary. Arbitrary dense distributions scale as $\mathcal{O}(N^2)$. However, such systems do not occur in standard MD simulations. It should be noted that other fast summation schemes (Barnes-Hut, Multigrid) suffer the same problem, since they are all based on a spatial domain decomposition. Schemes have been proposed [41] to overcome this problem. However, the additional overhead due to the tree management makes these schemes not favourable in most cases. A general discussion on the overall complexity of the FMM can be found in [42] and in [43]. It is stated that for machine precision calculations the complexity of the FMM is not $\mathcal{O}(N)$ but $\mathcal{O}(N \log^3 N)$ for a rotation-based approach and a homogeneous particle distribution. However, most simulations are carried out with a much lower precision.

A similar argument can be used to state that a direct summation with supposed $\mathcal{O}(N^2)$ complexity has an even worse complexity, since the precision decreases with increasing number of

Pass	Pass–Box Dependencies		Runtime	Percentage	Balanced
Setup*	tree, level-wise	$\downarrow\leftrightarrow$	4.12s	11.90%	—
Pass 1	tree (upwards)	$\uparrow$	1.87s	5.40%	—
Pass 2	level-wise	$\leftrightarrow$	13.32s	38.46%	•
Pass 3	tree (downwards)	$\downarrow$	0.72s	2.08%	—
Pass 4	lowest level	$\square$	2.60s	7.51%	—
Pass 5	neighbors	$\square\square$	12.00s	34.65%	•
Total			34.63s	100.00%	

* Includes full error control to define ws, d_{opt} and p_{opt} and the setup of the tree data structures.

Table 2.4.: Approximate percentage of computation time for every pass. The corresponding test system consists of 8^7 homogenously distributed particles. The error bound for the computation was set to $\Delta E_{\text{rel}} = 10^{-6}$. The runtime was optimized for pass 2 and pass 5. The symbols $\uparrow$ and $\downarrow$ depict operations up and down the FMM tree, the symbol $\leftrightarrow$ depicts operations in one level of the tree.

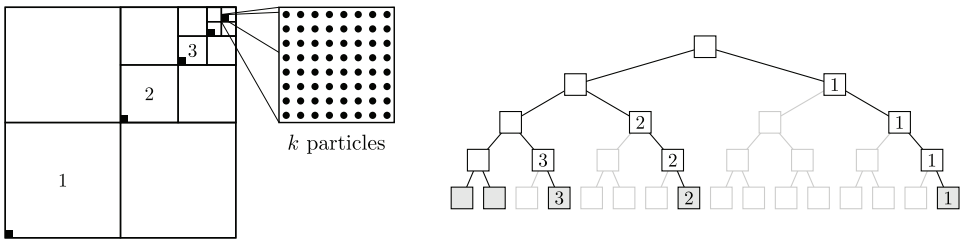
particles due to the number of executed operations. The numerical error [44] induced by the summation of N^2 pairs is statistically proportional to $\sqrt{N^2}\epsilon = N\epsilon$. For double precision calculations with machine precision $\epsilon = 2^{-52}$ and e.g. one billion particles, the result of the direct computation would yield only 7 significant digits instead of 16; 9 digits would be lost merely due to the summation. The dependency of the computation with respect to the number of particles therefore is not a special FMM problem, but inherent in the numerical summation of large number of terms. Compared to a direct summation the FMM will only lose $\sqrt{N}$ digits, since the scheme has a complexity of $\mathcal{O}(N)$.

Limited Order of Poles

Currently this implementation allows to simulate systems which can be represented with multipole moments up to 50. Numerical studies showed that for large homogeneous particle systems the order of poles did not exceed 50 poles not even for a requested relative energy precision of $\Delta E = 10^{-15}$. If the necessity arises the implementation can be extended to a higher order of poles.

Limited Number of Levels

The FMM tree is limited by the available integer number range on a certain machine. Most popular architectures nowadays provide 64-bit integers. Since we need 3 bits for each level it is possible to address 21 levels. Theoretically it is possible to store $\approx 10^{20}$ homogeneously distributed particles. However, the available memory does not allow such system sizes yet. For clustered distributions the 21-level limit can be reached much earlier, since an inhomogeneity



(a) Worst case particle distribution

(b) Corresponding binary tree

Figure 2.22.: The linear scaling of the FMM can only be achieved, if the tree depth grows $\log N$ with respect to the number of particles N. However, a worst case particle distribution (a) increases the complexity of the FMM to $\mathcal{O}(N^2)$. For each subset of k particles the FMM will set up an additional level in the tree (b).

demands deeper FMM trees. We easily reached an optimal FMM tree level of about 15 with only 100.000 heavily clustered particles.

Error Control for $ws > 1$

The FMM allows to set the well-separateness criterion ws to arbitrary values ($ws \geq 1$). However, the quadratical complexity overhead from the near field interactions makes this choice unfavourable. Numerical tests showed that for any tested particle system (homogeneous, clustered) a value of $ws = 2$ never achieved a better performance compared to the minimal value $ws = 1$ even for high precision calculations. Therefore, the error control scheme is limited to $ws = 1$, because the theoretical crossover of the order of poles and a larger near field is presumed to occur beyond 50 poles.

2.7. Error Analysis

The control of the approximation errors is a crucial part for any fast summation algorithm. Without error control the computation may lead to an unnecessary computational overhead if the error bound is rough or even worse to inaccurate results if no error bound exists at all. The well known cut-off scheme is afflicted with such issues. A small cut-off will result in a fast computation, but the results may be defective. A large cut-off might be too precise and will lead to a slow computation. Such an algorithm is not reliable for most applications and needs to be tuned for the intended purpose by the user himself. An ideal fast summation scheme would offer tight error bounds together with a minimal computation time for any given precision.

For the FMM there exists an *a priori* error bound. However, that error bound assumes a worst case scenario. This leads to a slow computation, since the worst case does not reflect the real

particle distribution and consistently overestimates the order of poles in the multipole expansion. In addition to the occurring truncation error, the error from the finite $\mathcal{B}$ translation has to be taken into account. Hence, the overestimation will increase even further.

In this section we want to derive an improved error bound with a smaller upper limit compared to the worst case scheme. Unlike the original approach this scheme will include all FMM related approximation errors and does depend on the particle distribution.

2.7.1. Worst Case Error Bound

To derive a valid error bound, we first have to revisit the three FMM parameters controlling the error. First, multipole and Taylor-like expansions are truncated at p poles. Thus, for a given p the FMM will impose two sources of error. The first source arises from the finite order of poles p we use in the expansions. The second source results from the use of operator $\mathcal{B}$. Both errors have to be considered in order to establish a thorough error control scheme. Despite the truncated expansions ω_{lm} the FMM parameter ws will affect the precision of the computation as well. The convergence of the expansion will improve rapidly for higher values of ws. Moreover, the number of levels L_{max} in the FMM tree influences the precision, because it defines the minimum size d_{min} of the smallest possible box in the tree. Besides the FMM parameters, properties of the particle system such as charge and particle distribution will influence the accuracy as well.

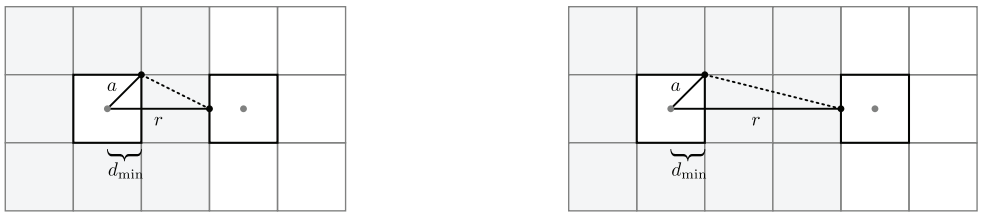
Truncation Error

First, let us consider the error for a given set (p, L_{max}, ws) neglecting the operator errors [26]. The approximated potential Φ_{FMM} introduces an error which depends on the positions of the particles inside their respective boxes. An upper bound can be established by

$$\begin{aligned}|\Phi_{\text{FMM}}(\mathbf{r}) - \Phi_{\text{exact}}(\mathbf{r})| &= |q| \cdot \left| \sum_{l=p+1}^{\infty} P_l(\cos\gamma) \frac{a^l}{r^{l+1}} \right| \\ &\leq \frac{|q|}{r-a} \left(\frac{a}{r}\right)^{p+1}.\end{aligned}$$

For a distribution-independent error bound, we have to assume the worst case distribution for the particles to obtain the largest error. Therefore, the ratio a/r has to be maximized. We must consider a position in box A which is located in one of the corners, with $a = \sqrt{3} \cdot d_{\text{min}}$. The position of the second particle in box B will be at the face of the box closest to box A with $r = (2ws + 1)d_{\text{min}}$ shown in figure 2.23. Therefore, the error in the potential $\Phi(\mathbf{r})$ for arbitrary ws is bounded by

$$|\Phi_{\text{FMM}}(\mathbf{r}) - \Phi_{\text{exact}}(\mathbf{r})| \leq \frac{\sum_{i=1}^{N} |q_i|}{(2ws + 1 - \sqrt{3})d_{\text{min}}} \left(\frac{\sqrt{3}}{2ws+1}\right)^{p+1}.$$



(a) $ws = 1$, $a = \sqrt{3} \cdot d_{\text{min}}$, $r = 3 \cdot d_{\text{min}}$ (b) $ws = 2$, $a = \sqrt{3} \cdot d_{\text{min}}$, $r = 5 \cdot d_{\text{min}}$

Figure 2.23.: To obtain an error bound for the truncation of the multipole expansion, we locate two particles at the least favourable positions inside box A and B. The chosen positions maximize the ratio a/r. The error bound depends on the choice of the separation criterion ws.

Translation Error

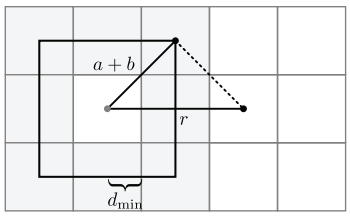
Additionally, the error connected with operator $\mathcal{B}$ has to be taken into account. By contrast, the operators $\mathcal{A}$ and $\mathcal{C}$ are exact up to the order of poles p and therefore do not need to be considered in an error estimation scheme. However, the truncation of the infinite sum of operator $\mathcal{B}$ will not be exact for $l \leq p$. Unfortunately, it is difficult to obtain a bound directly. With the help of operator $\mathcal{A}$, we are able to combine the bipolar expansion [45, 46] into a single expansion O_{lm}. Now we can use the same approach as in the last section. The bipolar expansion reduces to

$$\begin{aligned}\frac{1}{|\mathbf{r} - (\mathbf{a} + \mathbf{b})|} &= \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \sum_{j=0}^{l} \sum_{k=-j}^{j} O_{lm}(\mathbf{a}) M_{lm}(\mathbf{r}) O_{l-j,m-k}(\mathbf{b}) \\ &= \sum_{l=0}^{\infty} \sum_{m=-l}^{l} O_{lm}(\mathbf{a} + \mathbf{b}) M_{lm}(\mathbf{r}) \,.\end{aligned}$$

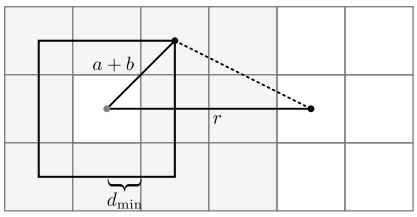
The new equation has a combined multipole at $\mathbf{a} + \mathbf{b}$ and a $\mathcal{B}$ operator with distance $\mathbf{r}$. The maximum value of $|\mathbf{a} + \mathbf{b}|$ is two times the worst case for two separated boxes , i.e. $2\sqrt{3} \cdot d_{\text{min}}$, and the minimum value for the box–box distance r is $2(ws + 1) \cdot d_{\text{min}}$ (see figure 2.24). Now the error bound can be specified by

$$|\Phi_{\text{FMM}}(\mathbf{r}) - \Phi_{\text{exact}}(\mathbf{r})| \leq \frac{\sum_{i=1}^{N} |q_i|}{(ws + 1 - \sqrt{3}) 2 d_{\text{min}}} \left(\frac{\sqrt{3}}{ws + 1} \right)^{p+1} .$$

Different worst case error bounds have been proposed [47] to reduce the order of poles. However, the worst case does not reflect the true particle distribution for the majority of particle systems and is therefore not practical. The overestimation due to the included translation error will further increase the bound. Some authors proposed schemes where the worst case is replaced by an empirical bound [48]. Others applied Chebyshev economizations [49] in order to reduce the order of poles p.



(a) $ws = 1$, $a+b = 2\sqrt{3}\cdot d_{\text{min}}$, $r = 4\cdot d_{\text{min}}$

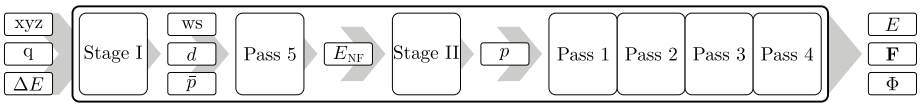


(b) $ws = 2$, $a+b = 2\sqrt{3}\cdot d_{\text{min}}$, $r = 6\cdot d_{\text{min}}$

Figure 2.24.: In addition the truncation error, the use of operator $\mathcal{B}$ induces errors. To obtain an upper bound we apply operator $\mathcal{A}$ to the bipolar expansion. Now the translation error can be obtained similarly to the truncation error. Again, the error bound depends on the separation criterion.

2.7.2. Improved Workflow / Error Bound

We established a new error scheme in order to introduce an improved *a priori* error bound with respect to the particle distribution. We do not take the error in the potential Φ as a measure, but the error in the total energy E of the system, since most applications require energy error bounds. However, the scheme can be derived for the potential or the forces as well. Let us start by specifying the modified workflow including the new error scheme:



1. Apply stage I of error control scheme. Utilize user-requested energy error ΔE to define the FMM parameter and set up FMM tree
   - Separation criterion ws
   - Tree depth d
   - Intermediate order of poles $\bar{p}_{\text{min}}$ assuming a homogeneous particle distribution inside each box on each level
2. Pass 5 ▷ Compute the near field contributions of the system
   - near field potential $\Phi_{\text{NF}}(x, y, z)$
   - near field forces $\mathbf{F}_{\text{NF}}(x, y, z)$
   - near field energy E_{NF}
3. Apply stage II of error control scheme to refine order of poles p
4. Pass 1 ▷ Expand and translate multipoles ω_{lm} up the tree
5. Pass 2 ▷ Transform multipole moments ω_{lm} into Taylor-like moments μ_{lm}
6. Pass 3 ▷ Translate Taylor-like moments μ_{lm} down the tree
7. Pass 4 ▷ Compute the far field contributions of the system

- far field potential $\Phi_{\text{FF}}(x, y, z)$
- far field forces $\mathbf{F}_{\text{FF}}(x, y, z)$
- far field energy E_{FF}

A separate error control scheme now precedes the original FMM scheme to obtain a set of optimal FMM parameters. The error scheme itself consists of two stages. After obtaining the set of parameters necessary to run the computation, the original FMM scheme continues in a slightly changed order. To obtain a precise near field energy in stage II, pass 5 is moved to the beginning of the computation. This rearrangement of pass 5 does not influence the far field computations in pass 1-4, but allows to use the exact near field energy to determine the final order of poles p in stage II. We will now derive both stages of the error estimation scheme.

2.7.3. Stage I of the Error Estimation Scheme

In the first stage of the error control scheme we assume a homogeneous particle distribution inside each box. In the second stage the real particle positions will be taken into account. This section briefly summarizes the basic ideas of the error control. A full derivation can be found in a recent publication by Dachsel [50].

Box-Box Energy

We start with the formula for the far field energy of two boxes with distance $\mathbf{R}$. Box A contains N_A particles, box B contains N_B particles. The multipole expansions for both boxes are described as $\omega_{lm}(\mathbf{a}_1)$ for box A and $\omega_{jk}(\mathbf{a}_2)$ for box B. The total energy reads

$$E = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \sum_{j=0}^{\infty} \sum_{k=-j}^{j} (-1)^j \omega_{lm}(\mathbf{a}_1) \frac{1}{R^{j+l+1}} \tilde{\tilde{P}}_{j+l,k+m}(\cos\theta) e^{i(k+m)\phi} \omega_{jk}(\mathbf{a}_2)\,. \qquad (2.11)$$

The multipole expansion for box A and B is described by

$$\omega_{lm} = \sum_{j=1}^{N} q_j a_j^l \tilde{P}_{lm}(\cos\alpha_j) e^{-im\beta_j}\,.$$

Averaged Multipole Moments

We assume the simulation box is extended from $[0, 1]$ in each spatial direction. If the coordinates of the considered system are not inside the cube with dimension $[0, 1] \times [0, 1] \times [0, 1]$, we scale the coordinates inside the $[0, 1]$-box and rescale after the computation accordingly. To simplify (2.11), we bound the phase factors of the multipole expansions and the $\mathcal{B}$ operator to their upper limit

$$|e^{-im\beta}| = 1, \quad |e^{i(k+m)\phi}| = 1\,.$$

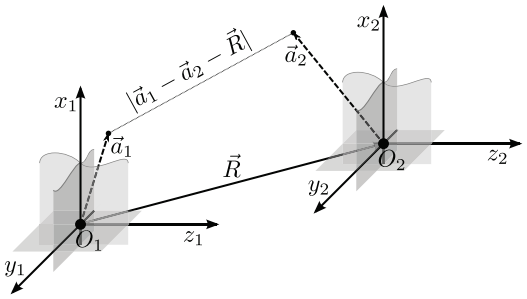


Figure 2.25.: Equation (2.11) is only valid, if we use the corresponding coordinate systems for the multipole expansions $\omega_{lm}(\mathbf{a}_1)$ and $\omega_{lm}(\mathbf{a}_2)$. The equation demands expansion 1 to be in the coordinate system O_1 and expansion 2 in coordinate system O_2. Interchanging the coordinate systems would yield an additional factor $(-1)^{j+l}$ in the energy.

We now define averaged multipole moments $\bar{\omega}_{lm}$ for a box with coordinate range $[-1, 1]$ as follows

$$\bar{\omega}_{lm} = \left(\sum_{i=1}^{N} |q_i| \right) \frac{\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} a^l \left| \tilde{P}_{lm}(\cos\alpha) \right| \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z}{\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z} \, .$$

The last equation can be simplified due to symmetry considerations

$$\bar{\omega}_{lm} = \left(\sum_{i=1}^{N} |q_i| \right) \int_{0}^{1}\int_{0}^{1}\int_{0}^{1} a^l \left| \tilde{P}_{lm}(\cos\alpha) \right| \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z \, .$$

A chargeless version $\bar{O}_{lm}$ of the average multipole $\bar{\omega}_{lm}$ can be defined as

$$\bar{O}_{lm} = \int_{0}^{1}\int_{0}^{1}\int_{0}^{1} a^l \left| \tilde{P}_{lm}(\cos\alpha) \right| \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z$$

and links to the charged version with

$$\bar{\omega}_{lm} = \left(\sum_{i=1}^{N} |q_i| \right) \bar{O}_{lm} \, .$$

The chargeless averaged multipoles $\bar{O}_{lm}$ are calculated numerically up to order $p = 250$ and are stored as constants for subsequent use. Additionally to the averaged multipoles we introduce a corresponding correction term

$$\frac{(j+l)!}{j!\, l!\, 2^{j+l+1}}$$

to compensate the overestimation of poles due to taking the absolute value of $\tilde{P}_{lm}$ in the integral. The compensation is used for all terms in the expansion except the monopole-monopole terms. For monopole interactions the associated Legendre polynomial $P_{00}(\cos\alpha)$ in the multipole moment ω_{00} as well as the associated Legendre polynomial $P_{00}(\cos\theta)$ for operator $\mathcal{B}$ are by definition

$$P_{00}(\cos\theta) = P_{00}(\cos\alpha) = 1$$

and therefore no correction is required. We will show later on that this additional term still maintains the error bound.

Define a Level-Independent $\mathcal{B}$ Operator

We now return to the expression of the box-box energy in (2.11). As depicted in figure 2.24, the distance R represents the distance of the box centers from A to B. However, the distance R between two arbitrary boxes in an interaction set depends on the actual FMM level L. A level-independent distance formulation with $d_{\text{min}} = 1$ yields

$$\hat{R} = 2^L R\,.$$

Now the distance $\hat{R}$ only depends on the separation criterion ws. The range of values for $\hat{R}$ is given by

$$2(ws+1) \leq \hat{R} \leq 2(2ws+1)\sqrt{3}$$

in case separated boxes exist on the actual level. The averaged chargeless version of the multipole expansions allows us to derive a corresponding box-box energy as follows

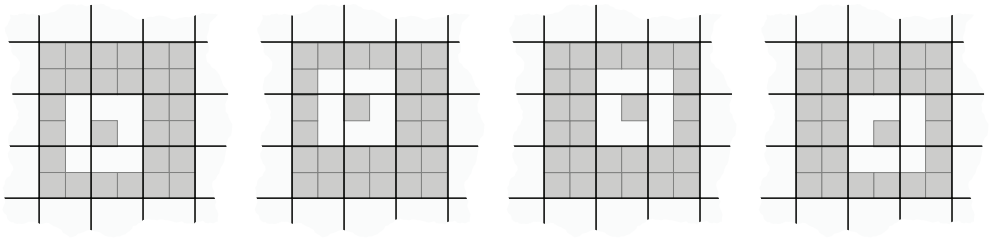
$$\bar{E} = \frac{1}{\hat{R}} + \sum_{l=0}^{\infty}\sum_{\substack{m=-l \\ j+l>0}}^{l}\sum_{j=0}^{\infty}\sum_{k=-j}^{j} \bar{O}_{lm}\frac{(j+l)!}{j!\,l!\,2^{j+l+1}}\left(\frac{1}{\hat{R}}\right)^{j+l+1}\left|P_{j+l,|m+k|}(\cos\theta)\right|\bar{O}_{jk}\,.$$

Since we do not want to include the correction term for the monopole we split the summation into a monopole term $1/\hat{R}$ and the remaining terms with $j+l>0$.

Averaging Over a Full Far Field Interaction Set

The derived energy formula still depends on the actual box-box distance. It can be further simplified by averaging over all box-box interactions in the interaction set for a given box. The number of boxes n in the interaction list depends on the separation criterion ws as follows

$$n_p = \underbrace{[2(2ws+1)]^3}_{\text{Full far field}} - \underbrace{(2ws+1)^3}_{\text{Near field}} = \underbrace{7(2ws+1)^3}_{\text{Interaction set}}\,.$$



(a) Child Box 1 (b) Child Box 2 (c) Child Box 3 (d) Child Box 4

Figure 2.26.: To obtain a fully symmetric interaction set, we average over all interaction sets of all 8 children of one parent box.

To avoid any effects from the asymmetric interaction set shown in figure 2.26, we also include all interactions from all 8 boxes in the same parent box. Therefore, the number of total interactions n_p increases by a factor of eight to

$$n_p = 56(2ws+1)^3 \,.$$

The averaged chargeless energy $\bar{\bar{E}}$ now reads

$$\bar{\bar{E}} = \frac{1}{n_p} \sum_{i=1}^{n_p} \bar{E} \,.$$

We change the order of summation and obtain

$$\begin{aligned} \bar{\bar{E}} =& \frac{1}{n_p} \sum_{i=1}^{n_p} \frac{1}{\hat{R}} + \\ & \frac{1}{n_p} \sum_{l=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=-l}^{l} \sum_{k=-j}^{j} \sum_{i=1}^{n_p} \bar{O}_{lm} \frac{(j+l)!}{j!\, l!\, 2^{j+l+1}} \left(\frac{1}{\hat{R}}\right)^{j+l+1} \left| P_{j+l,|m+k|}(\cos\theta) \right| \bar{O}_{jk} \end{aligned}$$

with $j + l > 0$.

Definition of Energy and Energy Error

Next, we introduce the element g_{00} for the first term

$$g_{00} = \frac{1}{n_p} \sum_{i=1}^{n_p} \frac{1}{\hat{R}}$$

and elements g_{jl} with $j + l > 0$ for the inner sums in the second term as

$$g_{jl} = \sum_{m=-l}^{l} \sum_{k=-j}^{j} \sum_{i=1}^{n_p} \bar{O}_{lm} \frac{(j+l)!}{j!\, l!\, 2^{j+l+1}} \left(\frac{1}{\hat{R}}\right)^{j+l+1} \left|P_{j+l,|m+k|}(\cos\theta)\right| \bar{O}_{jk}\,.$$

The energy $\bar{\bar{E}}$ is then found to be

$$\bar{\bar{E}} = g_{00} + \sum_{l=0}^{\infty} \sum_{j=0}^{\infty} g_{jl} \quad \text{with} \quad j + l > 0\,.$$

However, we need to introduce a truncation to proceed further, since an FMM implementation demands a finite order of poles p. Therefore, we split the infinite sums in the last equation, also displayed in figure 2.27, as follows

$$\begin{aligned} \bar{\bar{E}} &= g_{00} + \sum_{l=0}^{\infty} \sum_{j=0}^{\infty} g_{jl} \\ &= \underbrace{g_{00} + \sum_{l=0}^{p} \sum_{j=0}^{p} g_{jl}}_{\text{FMM Energy}} + \underbrace{\sum_{l=0}^{p} \sum_{j=p+1}^{\infty} g_{jl}}_{\text{FMM Error}} + \underbrace{\sum_{l=p+1}^{\infty} \sum_{j=0}^{p} g_{jl}}_{\text{FMM Error}} + \underbrace{\sum_{l=p+1}^{\infty} \sum_{j=p+1}^{\infty} g_{jl}}_{\text{FMM Error}} \\ &= \quad \bar{\bar{E}}(p) \quad + \quad \underbrace{\Delta\bar{\bar{E}}_{A1}(p) \;+\; \Delta\bar{\bar{E}}_{A2}(p) \;+\; \Delta\bar{\bar{E}}_{B}(p)}_{\Delta\bar{\bar{E}}(p)} \end{aligned} \tag{2.12}$$

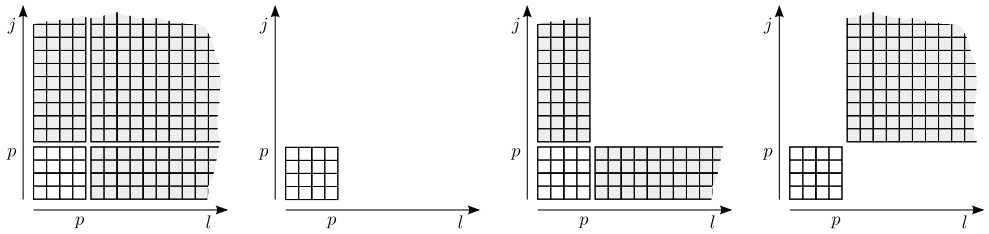
with $j + l > 0$ and obtain a suitable representation for our implementation. The first sum now represents the approximated FMM energy $\bar{\bar{E}}(p)$ up to p poles. The remaining sums represent the error $\Delta\bar{\bar{E}}(p)$ and can be combined to

$$\begin{aligned} \Delta\bar{\bar{E}}(p) &= \sum_{l=0}^{p} \sum_{j=p+1}^{\infty} g_{jl} + \sum_{l=p+1}^{\infty} \sum_{j=0}^{p} g_{jl} + \sum_{l=p+1}^{\infty} \sum_{j=p+1}^{\infty} g_{jl} \\ &= \sum_{l=0}^{p} \sum_{j=p+1}^{\infty} g_{jl} + \sum_{l=p+1}^{\infty} \sum_{j=0}^{\infty} g_{jl}\,. \end{aligned}$$

As described in section 2.6.4 for homogeneous particle distributions 50 poles are sufficient to obtain accuracies up to machine precision (double precision). Therefore, we can precompute and store the vector $\Delta\bar{\bar{E}}(p)$ for all orders of poles in the range $p \in \{0, \ldots, 50\}$.

Usage of the First Stage of the Error Scheme

In a simulation the error scheme could be used as follows. Starting from the root node (simulation box), we repeatedly divide the box into its eight childboxes until we reach level $L = 3$.



(a) $E(p) + \Delta E(p)$ (b) $E(p)$ (c) $\Delta E_{A1}(p) + \Delta E_{A2}(p)$ (d) $\Delta E_B(p)$

Figure 2.27.: The terms in (2.12) correspond to a certain order of summation. All subfigures are labeled accordingly.

Now, there exist separated boxes which can be treated in the far field. We compute the total charge Q_A of a box A and the total charge Q_B of a box B in the interaction list of A with

$$Q_A = \sum_{i=1}^{N_A} |q_i| \,, \qquad Q_B = \sum_{i=1}^{N_B} |q_i| \,.$$

For box $A(L)$ on a certain level L and its interaction set of boxes $B(A)$, we want to calculate an order of poles sufficient to meet the user-requested error bound ΔE_{req}. Therefore, we have to consider all interactions for box A and its interaction set of boxes $B(A)$ via

$$\Delta E_{\text{bb}} \geq \Delta \bar{\bar{E}}(p) Q_A \sum_{B(A)} Q_{B(A)}$$

to obtain a certain box-box error ΔE_{bb}. For systems with more than two non-empty boxes we have to include all remaining boxes $A(L)$ with their interaction sets on a certain level L by

$$\Delta E_{\text{L}} \geq \frac{1}{2} \Delta \bar{\bar{E}}(p) \sum_{A(L)} Q_A \sum_{B(A)} Q_{B(A)} \,.$$

Since we included interactions from box A with B and again box B with A, the right-hand side has to be divided by two to compensate the duplicate contributions. Finally by including all interactions from all levels starting from the lowest level $L_{\text{min}} = \lfloor \log_2(ws + 1) \rfloor + 2$ up to the current level L_{cur} and applying the level-dependent scaling factor 2^{L-1}, we obtain an inequality for the absolute error of the energy

$$\Delta E_{\text{req}} \geq \frac{1}{2} \Delta \bar{\bar{E}}(p) \sum_{L_{\text{min}}}^{L_{\text{cur}}} 2^{L-1} \sum_{A(L)} Q_A \sum_{B(A)} Q_{B(A)} \,.$$

The order of poles p sufficient to satisfy the requested error is obtained by increasing p from zero until the condition is met. If the estimated computation time of the FMM is not optimal

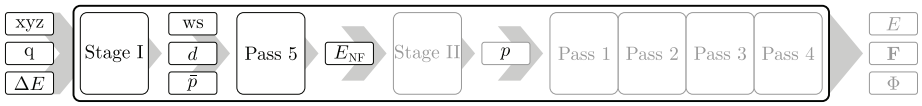
at the current level L_{cur}, the scheme will proceed and add a new level to the FMM tree. The procedure will continue until the optimal tree level L_{opt} is found or the maximum tree level L_{max} is reached.

2.7.4. Stage II of the Error Estimation Scheme

For homogeneously distributed particles, the first stage of the error control scheme is sufficient. However, realistic simulations often do not meet that demand of homogeneity. Therefore, an additional stage is required to guarantee the requested error bound ΔE_{req} also for inhomogeneous/clustered particle distributions.

Preliminary Steps

Similarly to the first stage, errors arising from box-box interactions are used to obtain an optimal error bound. Additionally, stage II takes into account the real particle distribution in a box with the following minor constraints. The real particle distribution is reduced to a particle distribution inside a fine grid with 33^3 grid points in box A and a particle distribution on the connecting lines from the center to the corners for box B. Let us recall the last steps we have performed to obtain the FMM parameter set. The following status applies:



Now we proceed with the second stage of the error scheme. Since the error of a single particle strongly depends on its position inside a box, we need to fix a certain tree depth before the second stage starts.

Simplification of the Interaction Set

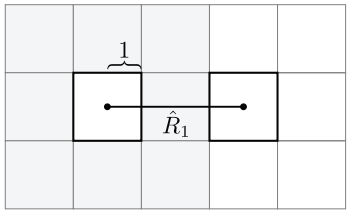
First, we have to reduce the number of possible box-box distances to a set of worst case box–box distances. We only incorporate the largest of the three distances d_x, d_y and d_z representing the distance in one dimension from the center of box A to the center of box B by

$$d_x = \left|x_c^A - x_c^B\right|, \qquad d_y = \left|y_c^A - y_c^B\right|, \qquad d_z = \left|z_c^A - z_c^B\right|$$

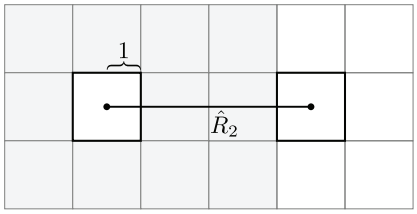
and from this

$$\hat{R} := d = \max(d_x, d_y, d_z) \leq \sqrt{d_x^2 + d_y^2 + d_z^2}\,.$$

Again we implicitly define a level-independent box-box distance $\hat{R}$ with $\hat{R} = 2^{L-1}R$. Depending on the chosen separation criterion ws, d can only take $(ws + 1)$ different values. The



(a) Minimum Box-Box Distance $\hat{R}_1 = 4$



(b) Maximum Box-Box Distance $\hat{R}_2 = 6$

Figure 2.28.: For a fixed separation criterion ws only two possible box-box distances exist for one spatial dimension.

distances $\hat{R}_s$ and the maximum distance $\hat{R}_{\text{max}}$ are defined by

$$\begin{aligned} \hat{R}_s &= 2(ws + s), \quad \text{with} \quad s \in \{1, \ldots, ws + 1\} \\ \hat{R}_{\text{max}} &= 2(2ws + 1)\,. \end{aligned}$$

For the considered case with $ws = 1$, only two possible distances exist

$$\hat{R}_1 = 4\,, \qquad \hat{R}_2 = \hat{R}_{\text{max}} = 6\,.$$

Positions of the Two Boxes

Without loss of generality, we assume a scenario which allows us to use the rotation-based FMM operators with $\mathcal{O}(p^3)$ complexity. Therefore, the two interacting boxes are assumed to be located along the z-axis with minimal distance $R := \hat{R}_1$ from box center A to box center B. The energy of such a system is obtained by

$$E = \sum_{l=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=-\min(j,l)}^{\min(j,l)} (\pm 1)^{j+l} (-1)^{j+m} \omega_{lm}(\mathbf{a}_1) \frac{(j+l)!}{R^{j+l+1}} \omega^*_{jm}(\mathbf{a}_2)\,. \tag{2.13}$$

Depending on the direction of interaction ($A \to B$ or $B \to A$), the factor $(-1)^{j+l}$ has to be used or dropped. Again we define a chargeless version of a multipole expansion O_{lm}, but since we want do take into account the real particle positions, we do not average, so that

$$O_{lm}(\mathbf{a}) = a^l \tilde{P}_{lm}(\cos\alpha) e^{-im\beta}\,.$$

Now we can replace the multipole moments in (2.13) with the chargeless version of box A with $O_{lm}(\mathbf{a}_1)$ and box B with $O_{jm}(\mathbf{a}_2)$ and rearrange the $\mathcal{B}$ operator. We obtain two energy formulas, depending on the relative position of the two boxes A and B depicted in figure 2.29.

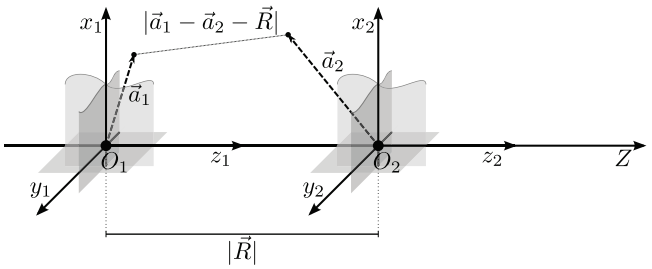


Figure 2.29.: The rotation-based FMM operators limit the possible box-box positions to the z-axis. Just like stage I the direction of the interaction matters. Interchanging the coordinate systems (or expansions) will result in an additional factor $(-1)^{j+l}$ in the energy (2.13).

The energies read

$$E_{\text{A-B}} = \frac{1}{R}\sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\sum_{m=-\min(j,l)}^{\min(j,l)}(-1)^{j+m}O_{lm}(\mathbf{a}_1)\frac{(j+l)!}{R^{j+l}}O^*_{jm}(\mathbf{a}_2)$$

$$E_{\text{B-A}} = \frac{1}{R}\sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\sum_{m=-\min(j,l)}^{\min(j,l)}(-1)^{l+m}O_{lm}(\mathbf{a}_2)\frac{(j+l)!}{R^{j+l}}O^*_{jm}(\mathbf{a}_1)\,.$$

We label the inner sum of the first equation as

$$\hat{S}_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) := \sum_{m=-\min(j,l)}^{\min(j,l)}(-1)^{j+m}O_{lm}(\mathbf{a}_1)\frac{(j+l)!}{R^{j+l}}O^*_{jm}(\mathbf{a}_2)$$

and due to symmetry the inner sum of the second equation

$$(-1)^{j+l}\hat{S}_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R}) = \sum_{m=-\min(j,l)}^{\min(j,l)}(-1)^{j+m}O_{lm}(\mathbf{a}_2)\frac{(j+l)!}{R^{j+l}}O^*_{jm}(\mathbf{a}_1)\,.$$

Since we do not want to store any information depending on the relative position of $\mathbf{a}_1$ and $\mathbf{a}_2$, we introduce a general position-independent S_{lj} as

$$S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) = (\pm 1)^{j+l}\hat{S}_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R})\,.$$

Following the scheme from the first stage, we now split up the outer infinite sums as follows

$$\sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\ldots = \sum_{l=0}^{p}\sum_{j=0}^{p}\ldots + \sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\ldots + \sum_{l=p+1}^{\infty}\sum_{j=0}^{p}\ldots + \sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty}\ldots\,.$$

The indices of the term

$$\sum_{l=p+1}^{\infty}\sum_{j=0}^{p} S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R})$$

can be swapped and the summation order is changed by applying the symmetry relation $O_{l,-m} = (-1)^m O_{lm}^*$, yielding

$$\sum_{l=0}^{p}\sum_{j=p+1}^{\infty} S_{jl}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) = \sum_{l=0}^{p}\sum_{j=p+1}^{\infty} (-1)^{j+l} S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})\,.$$

Hence, some error terms from the interaction of box A with box B are identical to error terms from the interaction from box B to box A. The energy for the interaction of box A with box B becomes

$$\begin{aligned}\Delta E_{\text{A-B}} =& \frac{1}{R}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) + (-1)^{j+l} S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})\Big] + \\ & \frac{1}{R}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty} S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R})\,.\end{aligned}$$

The energy for the interaction of box B with box A yields

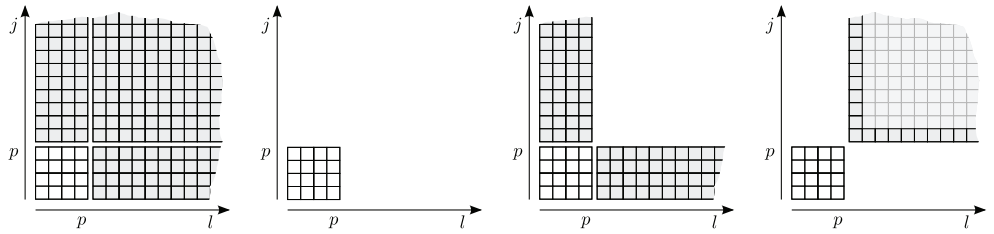
$$\begin{aligned}\Delta E_{\text{B-A}} =& \frac{1}{R}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[(-1)^{j+l} S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R}) + S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R})\Big] + \\ & \frac{1}{R}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty} (-1)^{j+l} S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})\,.\end{aligned}$$

By adding the two energies $E_{\text{A-B}}$ and $E_{\text{B-A}}$ we obtain

$$\begin{aligned}\Delta E =& \frac{1}{R}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) + (-1)^{j+l} S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})\Big] + \\ & \frac{1}{2R}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) + (-1)^{j+l} S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})\Big]\,.\end{aligned}$$

To simplify the last double sum, we only take terms with $j \geq p+1$ into account and drop all terms $l > p+2$ as shown in figure 2.30. The equation yields

$$\begin{aligned}\Delta E \approx& \frac{1}{R}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) + (-1)^{j+l} S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})\Big] + \\ & \frac{1}{2R}\sum_{j=p+1}^{\infty}\Big[S_{p+1,j}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) + (-1)^{j+l} S_{p+1,j}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})\Big]\,.\end{aligned} \tag{2.14}$$



(a) $E(p) + \Delta E(p)$ (b) $E(p)$ (c) $\Delta E_{l=\{0,\ldots,p\}}(p)$ (d) $\Delta E_{l=p+1}(p)$

Figure 2.30.: Similarly to stage I the infinite sums are split up. However, as shown in (d) only terms $l = p + 1$ are taken into account. All higher terms are neglected.

Now, both sums contain contributions $S_{lj}(\mathbf{a}_2, \mathbf{a}_1, \mathbf{R})$ expressed as interactions from box B on box A. We therefore neglect these terms in the following procedure, but we will account them in the scheme later on. We just have to guarantee that all box-box interactions occur. Therefore, we must perform interactions from box A and B and again the interaction from box B to A.

Following a similar scheme as proposed in the first stage of the error control we perform the summation of (2.14) stepwise with respect to l. For all values $l < p + 1$ only the first term of (2.14) needs to be considered. For the last term $l = p + 1$ only the last term of (2.14) has a contribution. Starting with the contribution from the first sum for $l = 0$ gives

$$\Delta E_{l=0} = \frac{1}{R} \sum_{j=p+1}^{\infty} S_{0j}(\mathbf{a}_1, \mathbf{a}_2, \mathbf{R}) \,. \tag{2.15}$$

It simplifies to

$$\begin{aligned} \Delta E_{l=0}(p) &= \frac{1}{R} \left| \sum_{j=p+1}^{\infty} (\pm 1)^j \frac{j!}{R^j} O_{j0}(\mathbf{a}_2) \right| \\ &= \frac{1}{R} \left| \sum_{j=p+1}^{\infty} (\pm 1)^j \left(\frac{a_2}{R}\right)^j P_{j0}(\cos\alpha_2) \right| . \end{aligned} \tag{2.16}$$

The term $(\pm 1)^{j+l}$ reduces to $(\pm 1)^j$, since we take the absolute value for each l separately. We call $\Delta E_{l=0}(p)$ the zeroth error contribution which is valid for any $p \geq 0$. We calculate all contributions from the zeroth order $l = 0$ up to $l = p + 1$.

We return to the original approach of the worst case error bound and compare the results. If we bounded the associated Legendre polynomial P_{j0} in (2.16) with its upper limit

$$\left|(\pm 1)^j P_{j0}(\cos\alpha_2)\right| \leq 1 \,,$$

the energy error would take a form similar to the original approach

$$\Delta E_{l=0} = \frac{1}{R - a_2} \left(\frac{a_2}{R}\right)^{p+1}.$$

A geometrical interpretation for the former equation can be drawn by assuming a worst case scenario with $d_{\text{min}} = 1$, maximum $a_2 = \sqrt{3}$ and minimum $R = \hat{R}_1 = 4$, yielding

$$\Delta E_{l=0} = \frac{1}{4 - \sqrt{3}} \left(\frac{\sqrt{3}}{4}\right)^{p+1}.$$

Compared to the worst case bound the error only varies in the value for R. Head-Gordon derived it from the expansion of the inverse distance in terms of Legendre polynomials. By contrast, our bipolar approach uses the full expansion in terms of the associated Legendre polynomials. Therefore, in the original approach the minimum distance of two particles is $R = 3$ and in our approach $R = 4$, since for $l = 0$ one particle is located at the center of the box. Again, treating only contributions from the zeroth term without taking into account the proper contributions from P_{j0} will result in an overestimation of poles. Therefore, we incorporate error terms of higher order and do not bound the elements P_{jk} to the upper limit.

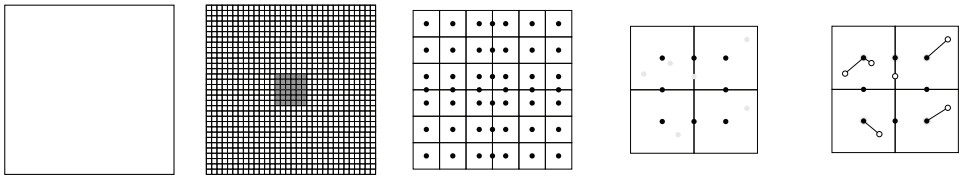
Establishing Dense Grid Points in Box A

Equation (2.16) still depends on the particle position and may take any arbitrary values inside box A. To reduce the number of possible particle positions, we introduce a fine grid with 32 uniform grid points per dimension in box A. Additional grid points are introduced on each coordinate axis yielding a total of $G = 33^3$ grid points in box A. All particles within one of the 32^3 grid boxes are mapped to the center of the grid box, i.e. the grid point. Particles with coordinates on one of the three axes are mapped onto the next grid point on the axis. Since the number of possible distances a_1 and angles α_1 and β_1 now depends on the underlying grid, we introduce new coordinates for box A as

$$\begin{aligned} a_G \quad &\text{with} \quad \lim_{G\to\infty} a_G = a_1\,, \\ \alpha_G \quad &\text{with} \quad \lim_{G\to\infty} \alpha_G = \alpha_1\,, \\ \beta_G \quad &\text{with} \quad \lim_{G\to\infty} \beta_G = \beta_1\,. \end{aligned}$$

Performing a Step-Wise Calculation of $\Delta E(p)$

Now we calculate the infinite sum in (2.16) for every pole up to a certain p numerically and store the calculated values. We reduce the storage requirement by only keeping the maximum value



(a) arbitrary subbox (b) 32^3 grid boxes (c) grid points (d) grid and particles (e) mapped particles

Figure 2.31.: Every box on every level is subdivided to apply a spatial grid with 33^3 grid points. All particles inside a grid box are mapped to the center of this grid box; the grid point. Particle on the coordinate axes are mapped to additional points on the axes.

for $(\pm 1)^j$ in

$$F_{0p}(a_G, \alpha_G) = \max_{s\in\{-1,1\}} \left| \sum_{j=p+1}^{\infty} s^j \left(\frac{a_G}{R}\right)^j P_{j0}(\cos\alpha_G) \right| .$$

Next we examine errors from the first order $l = 1$. The contribution of this term is much more complicated, since it depends on the coordinates of both particles. We write

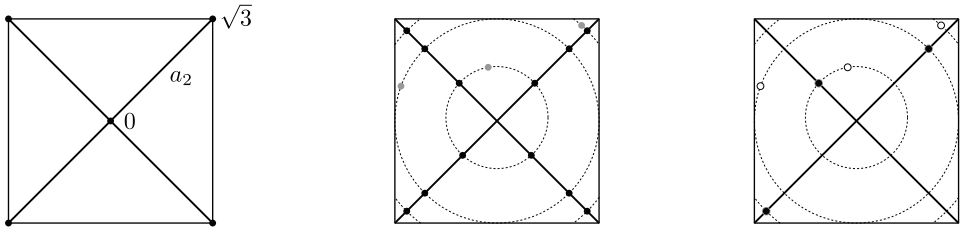
$$\Delta E_{l=1}(p) = \frac{1}{R} \sum_{m=-1}^{1} \sum_{j=p+1}^{\infty} S_{1j}(\mathbf{a}_G, \mathbf{a}_2, \mathbf{R}) \tag{2.17}$$

with

$$\begin{aligned}\Delta E_{l=1}(p) = &\frac{1}{R}\frac{a_1}{R} \Bigg| P_{10}(\cos\alpha_2) \sum_{j=p+1}^{\infty} (\pm 1)^j (j+1) \left(\frac{a_G}{R}\right)^j P_{j0}(\cos\alpha_G) - \\ &P_{11}(\cos\alpha_2)\cos(\beta_2 - \beta_G) \sum_{j=p+1}^{\infty} (\pm 1)^j \left(\frac{a_G}{R}\right)^j P_{j1}(\cos\alpha_G) \Bigg| .\end{aligned}$$

Defining Angular Parts of the Non-Grid Box B

Since we introduced a grid in box A, we should do the same for box B. However, an additional fine grid would increase the complexity of the error scheme to $\mathcal{O}(M_1 M_2)$ with M_1 being the number of particles in box A and M_2 being the number of particles in box B. Therefore, we introduce a different procedure to handle error terms with respect to the coordinates of the second box. The largest impact on the error from box B is caused by the radial part a_2 of the particle coordinates with respect to the center of box B. Therefore, we do not confine the range of values and allow all possible values $a_2 \in [0, \sqrt{3}]$. However, the angles α_2 and β_2 are treated differently.



(a) Non-grid Box $a_2 \in [0, \sqrt{3}]$ (b) Possible Angles (c) Mapping of Particles

Figure 2.32.: The second box allows all values for a_2, but limits the angles α_2 and β_2 to the connecting lines inside box B.

To ensure that the radial part can take its minimum $a_2^{\min} = 0$ and maximum value $a_2^{\max} = \sqrt{3}$, the available angles are thereby fixed. The minimum value for a_2 coincides with the center of the box. The maximum value for a_2 represents one of the 8 corners (see figure 2.32). Hence, we only allow angles α_2 and β_2 on the connecting lines from the center to all box corners. We introduce new coordinates for the angular part of box B so that

$$\begin{aligned} &\alpha_{2_n} \quad \text{with} \quad n \in \{1, \ldots, 8\} \quad \text{and} \quad \cos\alpha_{2_n} = \pm\frac{\sqrt{3}}{3} \\ &\beta_{2_n} \quad \text{with} \quad n \in \{1, \ldots, 8\} \quad \text{and} \quad \cos\beta_{2_n} = \pm\frac{\sqrt{2}}{2}\,,\ \sin\beta_{2_n} = \pm\frac{\sqrt{2}}{2}\,. \end{aligned}$$

To avoid storing the grid box A for every possible angular part in box B we only store the maximum error contribution for every grid point G and $\alpha_{2_n}, \beta_{2_n}$. Therefore, the elements of F_{1p} implicitly depend on the maximum of the angular part of box B. Finally, the error terms of the first order $\Delta E_{l=1}(p)$ are found to be

$$F_{1p}(a_G, \alpha_G, \beta_G) = \max_{n\in\{1,\ldots,8\}} \max_{s\in\{-1,1\}} \left|\widetilde{F}_{1p}^{n}(a_G, \alpha_G, \beta_G)\right|$$

with

$$\begin{aligned} \widetilde{F}_{1p}^{n}(a_G, \alpha_G, \beta_G) =& P_{10}(\cos\alpha_{2_n}) \sum_{j=p+1}^{\infty} s^j (j+1) \left(\frac{a_G}{R}\right)^j P_{j0}(\cos\alpha_G) - \\ & P_{11}(\cos\alpha_{2_n}) \cos(\beta_{2_n} - \beta_G) \sum_{j=p+1}^{\infty} s^j \left(\frac{a_G}{R}\right)^j P_{j1}(\cos\alpha_G)\,. \end{aligned}$$

Higher elements $\widetilde{F}_{2p}, \widetilde{F}_{3p}, \ldots \widetilde{F}_{qp}$ can be derived similarly. However, we will skip a somewhat tedious derivation and continue directly with the equation for the general element $\widetilde{F}_{qp}$ with

$q \leq p+1$. It can be written as follows

$$\widetilde{F}^n_{qp}(a_G, \alpha_G, \beta_G) = (2-\delta_{m0}) \sum_{m=0}^{q} (-1)^m \tilde{P}_{qm}(\cos\alpha_{2_n}) \cos(m\beta_{2_n} - m\beta_G) \times \sum_{j=p+1}^{\infty} s^j (j+q)! \left(\frac{a_G}{R}\right)^j \tilde{P}_{jm}(\cos\alpha_G) \,.$$

It should be noted that the terms $q = p+1$ are generated by the second double sum in (2.14). We only store the largest contribution for the error in F_{qp} as

$$F_{qp}(a_G, \alpha_G, \beta_G) = \frac{1}{1+\delta_{p+1-q,0}} \max_{n\in\{1,\ldots,8\}} \max_{s\in\{-1,1\}} \left| \widetilde{F}^n_{qp}(a_G, \alpha_G, \beta_G) \right| \,.$$

Symmetry Considerations

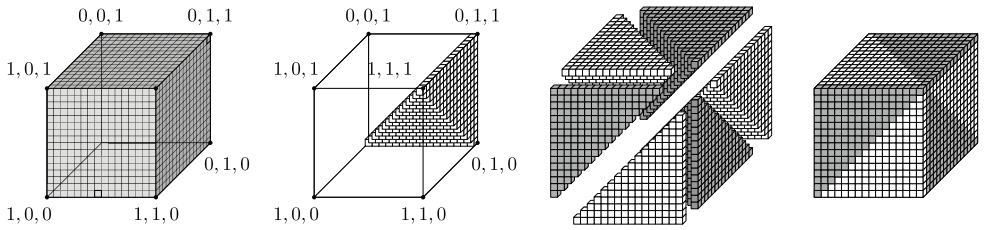
A general bipolar expansion consists of a set of six independent coordinate variables, three coordinates for particles within grid box A with a_G, α_G, β_G and three coordinates for particles within box B with a_2, α_2, β_2. We will reduce the set to four variables $a_2, a_G, \alpha_G, \beta_G$. The overall storage requirement for all elements F_{qp} with $0 \leq q \leq p+1 = 50$ would be $(p+1)\cdot(p+2)\cdot 33^3 \cdot 8$ Bytes, approximately 730 MBytes. To reduce the total amount of precomputed data, we utilize the symmetry inherent in the approach. By employing the minimal distance d in one dimension for all possible box–box distances, any rotations of the global coordinate system around the coordinate axes must lead to the identical result for the error terms. Therefore, we are able to reduce the number of grid points and benefit from a lower memory requirement. Representing the elements F_{qp} in Cartesian coordinates and using the mentioned symmetry relations implies

$$\begin{aligned} F_{qp}(x_G, y_G, z_G) &= F_{qp}(x_G, z_G, y_G) \\ &= F_{qp}(y_G, x_G, z_G) \\ &= F_{qp}(y_G, z_G, x_G) \\ &= F_{qp}(z_G, x_G, y_G) \\ &= F_{qp}(z_G, y_G, x_G) \,. \end{aligned}$$

The number of elements is reduced by a factor of six. Additionally we take the symmetry for $\pm x, \pm y, \pm z$ into account, since we cannot distinguish between interactions with a special direction by writing

$$F_{qp}(x_G, y_G, z_G) = F_{qp}(\pm x_G, \pm y_G, \pm z_G) \,.$$

Again the number of elements is reduced by a factor of 8. By now the total number of elements should be reduced by a factor of 48. Since some faces and edges share some elements, which



(a) Gridded Set, 1st Octant (b) Reduced Grid Set H_{qp} (c) Redundant Grid Boxes (d) Full Grid Set

Figure 2.33.: Only one octant of the full grid box $[-1, 1]$ is shown. Due to symmetry, the number of unique grid boxes can be reduced from 33^3 to 969. The worst case elements from all redundant grid points are stored in their reduced set of grid points.

can be seen in figure 2.33b, the set is reduced to 969 elements yielding a shrink factor of approximately 37. The error control demands to sum up contributions for all F_{0p} terms to achieve an upper bound. However, numerical tests showed that more than 99% of all possible interactions for a certain worst-case box-box configuration are sufficiently covered by the maximum term only. The remaining "error-prone" contributions (1%) only occur for

- 6 out of 189 box–box configurations,
- boxes with only few particles.

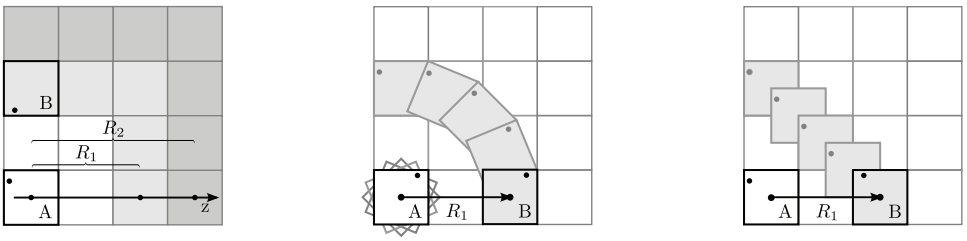
Since we add up error contributions from box–box interactions on all levels, the influence of the "error-prone" contributions can be neglected. Nevertheless additional measures have been implemented to maintain the error bound. Details can be found in [50].

Including the worst-case elements and store contributions H_{qp} separately we get

$$H_{0p}(x,y,z) = \max \left\{ \begin{array}{l} F_{0p}(x,y,z) \\ F_{0p}(x,z,y) \\ F_{0p}(y,x,z) \\ F_{0p}(y,z,x) \\ F_{0p}(z,x,y) \\ F_{0p}(z,y,x) \end{array} \right\}$$

$$\vdots$$

$$H_{qp}(x,y,z) = \max \left\{ \begin{array}{l} F_{qp}(x,y,z) \\ F_{qp}(x,z,y) \\ F_{qp}(y,x,z) \\ F_{qp}(y,z,x) \\ F_{qp}(z,x,y) \\ F_{qp}(z,y,x) \end{array} \right\}$$



(a) Original Positions (b) Rotated Positions (c) Shifted Positions

Figure 2.34.: The reduced set of grid boxes does not have any axis information stored in it. Therefore, we do not need to execute an expensive rotation to move an interacting box on the Z axis required in the approach. We simply shift the expansion. The worst-case error terms from the full set of grid points ensure an adequate handling.

with $x, y, z \in \left\{0, \pm\frac{1}{32}, \ldots, \pm\frac{29}{32}, \pm\frac{31}{32}\right\}$, $p \in \{0, \ldots 50\}$ and $q \in \{0, \ldots p+1\}$.

Radial Part of the Non-grid Box B

The error terms stored in H_{qp} incorporate five of six coordinate variables. The remaining radial part of particles in box B with a_2 is subject of this section. We calculate functions f_q for a set of six sampling points $q \in \{0, 4, 9, 18, 36, 45\}$ with the level-independent box–box distance $\hat{R}$ and N_B being the number of particles in box B as

$$f_q = \sum_{i=1}^{N_B} |q_i| \left(\frac{a_i}{\hat{R}}\right)^q .$$

The number of sampling points is chosen based on memory and accuracy considerations. A single sampling point is not sufficient to represent the complicated error terms. Dozens of sampling points help to increase accuracy, but cause a larger memory footprint. Therefore, we limit the number of points to six, which ensures a satisfactory balance. The first three sampling points are necessary to obtain a sufficient error representation for low precision calculations with only a few multipoles. The last three sampling points provide a basis for high precision calculations. The f_q have to be computed for each box on each level, similarly to the grid box A. We compute these elements separately for each of the $ws + 1$ displacements $s \in \{1, \ldots, ws + 1\}$. For the discussed case with $ws = 1$, we require two functions f_q^1 and f_q^2. The level-independent distance $\hat{R}$ becomes $\hat{R}_s = 2(ws + s)$. For a general f_{qp}^s we modify the last equation as follows

$$f_{qp}^s = \left(\frac{\hat{R}_1}{\hat{R}_s}\right)^{q+p+2} \sum_{i=1}^{N_B} |q_i| \left(\frac{a_i}{\hat{R}_1}\right)^q .$$

The term

$$\left(\frac{\hat{R}_1}{\hat{R}_s}\right)^{q+p+2} = \underbrace{\left(\frac{\hat{R}_1}{\hat{R}_s}\right)^{q}}_{l=q} \cdot \underbrace{\left(\frac{\hat{R}_1}{\hat{R}_s}\right)^{p+1}}_{j=p+1} \cdot \underbrace{\left(\frac{\hat{R}_1}{\hat{R}_s}\right)}_{\text{common factor}}$$

comprises all parts of $1/R$ from the general energy (2.13).

To link the error terms from the grid box and error terms from the radial part in the non-grid box, we set up a corresponding grid structure Q_{qp}^L similarly to the precomputed grid H_{qp}. The elements of the second grid are calculated at runtime. Q_{qp}^L stores all contributions from the functions f_{qp}^s for any non-empty grid box in box B. The level-independent definition of the box–box distance $\hat{R}$ allows us to sum over all contributions from all boxes. We define the Q_{qp}^L as

$$Q_{qp}^L(G) = 2 \sum_{A(L)} q_G \max_{A(B)} \max_{s\in\{1,2\}} f_{qp}^s \tag{2.18}$$

The last equation can be used in the error scheme as follows. For every grid-box G with grid-box charge q_G, $Q_{qp}^L(G)$ stores the error terms of the radial part a_2 of box B and the charges of both grid box G with $q_G = \sum_{i=1}^{N_G} |q_i|$ and non-grid box B. We do not sum up all error contributions from the interaction set $A(B)$, but take the maximum. Finally, we sum up all sets $A(L)$ for a certain level L. Hence, we store contributions from different levels L in different lists $Q_{qp}^L(G)$. All contributions from one level however are stored in an unique super grid box $Q_{qp}^L(G)$.

Interaction Sets

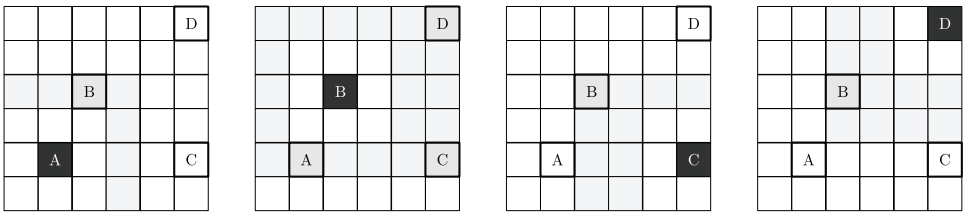
To obtain the full set of error terms in $Q_{qp}^L(G)$ we have to guarantee to have each box twice in the interaction set. This requirement would be satisfied by summing up all interactions from the interaction set $A(B)$. However, it is sufficient to store the contribution for the maximum error only. Since, this scheme does not guarantee to include interactions from box A with B and in reverse, we have to multiply by a factor of two to compensate for the possible asymmetry. The asymmetric interaction set is depicted in figure 2.35.

Interpolation Scheme for $Q_{qp}^L(G)$

The calculation of $Q_{qp}^L(G)$ is only performed for the 6 sampling points $q \in \{0, 4, 9, 18, 36, 45\}$. To obtain the remaining values for $q \in \{0, \ldots, 51\}$ we apply an interpolation scheme. We know

$$Q_{0,p}^L(G) > Q_{4,p}^L(G) > Q_{9,p}^L(G) > Q_{18,p}^L(G) > Q_{36,p}^L(G) > Q_{45,p}^L(G)$$

and since the function Q_{qp}^L is strictly monotonic decreasing, we use an interpolation scheme to obtain the missing values.



(a) $\max\{S_A\} = B$ (b) $\max\{S_B\} = C$ (c) $\max\{S_C\} = B$ (d) $\max\{S_D\} = B$

Figure 2.35.: The application of the maximum instead of the sum in (2.18) demands additional care, since we have to guarantee that the error terms from the individual box–box interaction are included. This figure shows that the maximum does not guarantee that the symmetry is conserved. Therefore, we add an additional factor of two to guarantee the error bound.

Linking Box A with Box B

The last sections described how we obtained the error terms for grid box A and the non-grid box separately. To obtain the error terms for the interaction of these two boxes, we multiply the contributions from A and B separately for every different order of poles q. We obtain

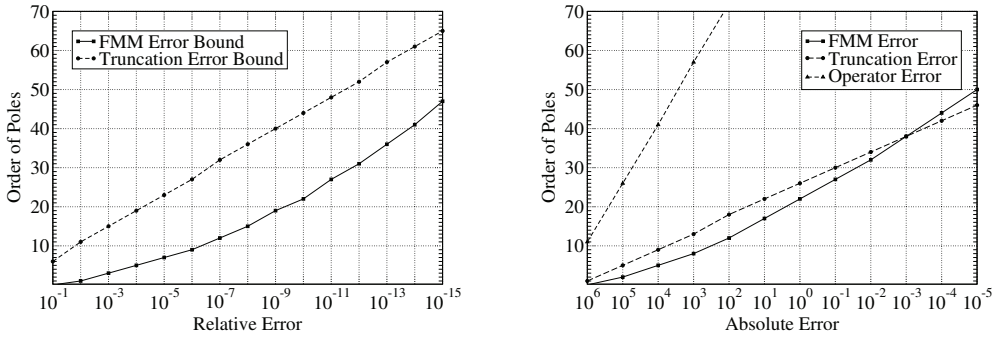
$$\begin{aligned} T_0^L(G) &= Q_{0p}^L(G) \cdot H_{0p}(G) \\ T_1^L(G) &= Q_{1p}^L(G) \cdot H_{1p}(G) \\ &\vdots \\ T_q^L(G) &= Q_{qp}^L(G) \cdot H_{qp}(G) \\ T_{q+1}^L(G) &= Q_{q+1,p}^L(G) \cdot H_{q+1,p}(G)\,. \end{aligned}$$

Finally, we take the maximum of the error terms T_q and rescale the level-independent distance. We sum up all error terms from all super grid boxes and all levels.

$$\Delta E_{\text{req}} \geq \sum_{L=L_{\text{min}}}^{L_{\text{opt}}} \sum_{G} \frac{2^{L-1}}{ws+1} \max \left\{ \begin{array}{c} T_0^L(G) + T_1^L(G) \\ T_2^L(G) \\ \vdots \\ T_q^L(G) \\ T_{q+1}^L(G) \end{array} \right\}. \tag{2.19}$$

To obtain the necessary order of poles p for a computation, we start with $q = 0$ and increase q until the right hand side of (2.19) is less or equal ΔE_{req}. The starting level L_{min} depends on the separation criterion ws with

$$L_{\text{min}} = \lfloor \log_2(ws+1) \rfloor + 2\,.$$

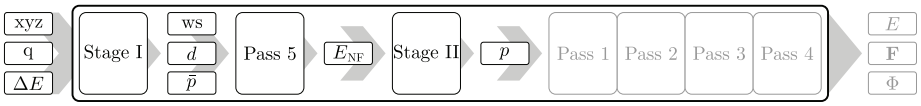


(a) 8^7 homogeneously distributed particles

(b) 10^5 particles on a line, $E = 1109014,612\ldots$

Figure 2.36.: The new error scheme significantly reduces the order of poles for a given precision (a). However, for inhomogeneous particle distributions the truncation error may under- or overestimate the order of poles (b). Therefore, any scheme only depending on the truncation error may not maintain the requested error bound.

After finalizing the second stage, the computation starts with the remaining FMM passes 1–4. The current status of the computation is as follows

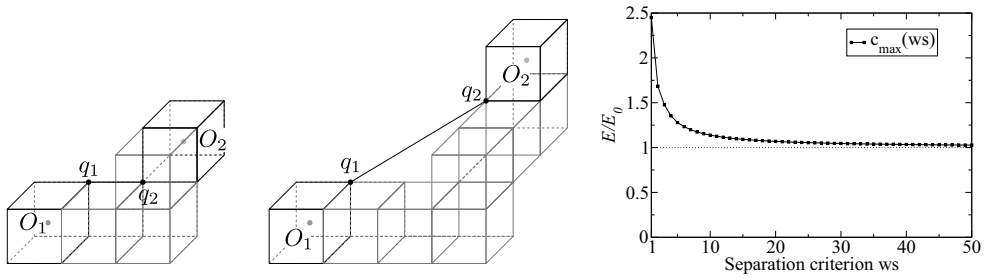


2.7.5. FMM without Far Field Contributions

The presented error scheme will return the order of poles $p \geq 0$ necessary to meet the requested error bound. In some cases the computation of the far field can be neglected at all for $p < 0$ since the energy obtained from the near field part already meets the error bound. In such a case the presented method reduces to a cut-off scheme and the examined particle system can be regarded as system without long-range interactions for the requested error bound. Especially for periodic systems presented in chapter 3, the computation time can be further reduced for very low precision calculations. We define an upper bound for the ratio of the exact far field energy E and the monopole energy E_0 by

$$\frac{E}{E_0} \leq c_{\max} .$$

For $ws = 1$ the maximal ratio is $\sqrt{6} : 1$. For $ws = 2$ the maximal ratio is $\sqrt{17} : 6$.



(a) $c_{\max}(ws = 1)$ (b) $c_{\max}(ws = 2)$ (c) $c_{\max}(ws = \{1 \dots 50\})$

Figure 2.37.: Upper bound $c_{\max}$ for different worst-case particle positions depending on the separation criterion ws. For infinite ws, $\lim_{ws\to\infty} c_{\max}(ws) = 1$.

2.7.6. Further Analysis of the Error Bound

Let us reconsider the first stage of the error control now following a slightly different approach. Again we start with the formula for the far field energy of two boxes with distance $\mathbf{R} = (R, \theta, \phi)$. The multipole expansions for both boxes are labeled as $\omega_{lm}(\mathbf{a}_1)$ for box A and $\omega_{jk}(\mathbf{a}_2)$ for box B. The total energy reads

$$E = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \sum_{j=0}^{\infty} \sum_{k=-j}^{j} (-1)^j \omega_{lm}(\mathbf{a}_1) \frac{1}{R^{j+l+1}} \tilde{\tilde{P}}_{j+l,k+m}(\cos\theta) e^{i(k+m)\phi} \omega_{jk}(\mathbf{a}_2)\,. \qquad (2.20)$$

The multipole expansion for a collection of particles N with spherical coordinates (a_j, α_j, β_j) is described by

$$\omega_{lm} = \sum_{j=1}^{N} q_j a_j^l \tilde{P}_{lm}(\cos\alpha_j) e^{-im\beta_j}\,.$$

In contrast to the original approach we do not bound the phase factors $e^{-im\beta_j}$ to one. Instead we reformulate the multipole expansion $\omega_{lm}(\mathbf{a})$ in Cartesian coordinates (x, y, z) [27]. Since we are going to average the expansion inside each box, we define a chargeless multipole expansion

$$O_{lm}^* = (-1)^m \sum_{k=0}^{\lfloor \frac{l-m}{2} \rfloor} \frac{(-1)^k (x+iy)^{k+m} (x-iy)^k z^{l-m-2k}}{2^{2k+m}(k+m)!k!(l-m-2k)!}\,.$$

Averaged Multipole Moments

Without loss of generality, we define averaged multipole moments $\bar{\omega}_{lm}$ for a box with coordinate range $[-1, 1]$ as follows

$$\bar{\omega}_{lm} = \left(\sum_{i=1}^{N} |q_i| \right) \frac{\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} O_{lm} \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z}{\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z} \, .$$

The former integral is almost identical to the integral introduced in section 2.7.3. Instead of using bounded phase factors, we solve the integral analytically. The last equation simplifies due to symmetry considerations to

$$\bar{\omega}_{lm} = \left(\sum_{i=1}^{N} |q_i| \right) \int_0^1\int_0^1\int_0^1 O_{lm} \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z \, .$$

A chargeless version $\bar{O}_{lm}$ of the average multipole $\bar{\omega}_{lm}$ can be defined as

$$\bar{O}_{lm} = \int_0^1\int_0^1\int_0^1 O_{lm} \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z$$

and links to the charged version with

$$\bar{\omega}_{lm} = \left(\sum_{i=1}^{N} |q_i| \right) \bar{O}_{lm} \, .$$

Since our chargeless multipole expansion O_{lm} is formulated in Cartesian coordinates, we solve the integral analytically for each l and m up to a maximum truncation order of 50. The integration was performed with the help of the computer algebra system MATHEMATICA [51]. A table containing the first 20 non-zero elements can be found in Appendix A.1.

Level-Independent $\mathcal{B}$ Operator

To compute the box-box interaction energy with regard to (2.20) we also have to compute the M2L operator for all possible box-box interactions. The number of box-box interactions is limited to

$$n_p = \underbrace{[2(2ws+1)]^3}_{\text{Full far field}} - \underbrace{(2ws+1)^3}_{\text{Near field}} = \underbrace{7(2ws+1)^3}_{\text{Interaction set}} \, .$$

For $ws = 1$ we have a total of 189 box-box pairs. Due to symmetry considerations only 16 different sets can be identified. Therefore, we have to compute 16 different M2L operators to

cover all 189 box-box pairs. Following our original scheme we expand all M2L operators up to $2p = 100$.

We define the energy of two interacting boxes with averaged multipoles as follows

$$\bar{E} = \bar{E}(p) + \Delta\bar{E}(p)$$

with

$$\bar{E}(p) = \sum_{l=0}^{p} \sum_{m=-l}^{l} \sum_{j=0}^{p} \sum_{k=-j}^{j} (-1)^j \bar{\omega}_{lm}(\mathbf{a}_1) \frac{1}{R^{j+l+1}} \tilde{\tilde{P}}_{j+l,k+m}(\cos\theta) e^{i(k+m)\phi} \bar{\omega}_{jk}(\mathbf{a}_2)\,.$$

In the original approach we precomputed the error vector $\Delta\bar{E}(p)$ for all order of poles in the range $p \in \{0, \ldots, 50\}$. Now we want to derive a different approach. First, we calculate the exact interaction energy $\bar{E}$. Afterwards, we simply compute for all p in the range $p \in \{0, \ldots, 50\}$

$$\Delta\bar{E} = \bar{E} - \bar{E}(p)\,.$$

Analytic Solution for the Interaction Energy $\bar{E}$

The evaluation of the Coulomb integral for cuboids with a uniform (averaged) density was presented in an analytic form by Mura and Handy in [52]. We will now derive a scheme to compute the exact interaction energy for a box-box interaction in the FMM. We restrict ourselves to the case $ws = 1$, but the scheme may as well be used for any other separation criterion ws.

The energy for a cuboid of size $J \times K \times L$ with uniform density is given by

$$E_{JKL} = \int_0^J \int_0^J \int_0^K \int_0^K \int_0^L \int_0^L \frac{\mathrm{d}x_1\,\mathrm{d}x_2\,\mathrm{d}y_1\,\mathrm{d}y_2\,\mathrm{d}z_1\,\mathrm{d}z_2}{\sqrt{(x_1-x_2)^2 + (y_1-y_2)^2 + (z_1-z_2)^2}}\,.$$

The solution presented by Mura and Handy can be summarized as

$$\bar{E}_{JKL} = \frac{1}{30}\bar{E}'_{JLK} \tag{2.21}$$

with

$$\begin{aligned}
\bar{E}'_{JKL} =& 5J^4K\log\left(\frac{\sqrt{K^2+J^2}+K}{\sqrt{K^2+J^2}-K}\right)+5KL^4\log\left(\frac{\sqrt{L^2+K^2}+K}{\sqrt{L^2+K^2}-K}\right)+\\
& 5JK^4\log\left(\frac{\sqrt{K^2+J^2}+J}{\sqrt{K^2+J^2}-J}\right)+5J^4L\log\left(\frac{\sqrt{L^2+J^2}+L}{\sqrt{L^2+J^2}-L}\right)+\\
& 5JL^4\log\left(\frac{\sqrt{L^2+J^2}+J}{\sqrt{L^2+J^2}-J}\right)+5K^4L\log\left(\frac{\sqrt{L^2+K^2}+L}{\sqrt{L^2+K^2}-L}\right)-\\
& 5L(K^4-6J^2K^2+J^4)\log\left(\frac{R+L}{R-L}\right)-\\
& 5K(L^4-6J^2L^2+J^4)\log\left(\frac{R+K}{R-K}\right)-\\
& 5J(L^4-6K^2L^2+K^4)\log\left(\frac{R+J}{R-J}\right)+\\
& 40JKL(J^2-K^2)\sin^{-1}\left(\frac{JL}{\sqrt{K^2+J^2}\sqrt{L^2+K^2}}\right)+\\
& 40JKL(J^2-L^2)\sin^{-1}\left(\frac{JK}{\sqrt{L^2+J^2}\sqrt{L^2+K^2}}\right)+\\
& 4(L^4-3(K^2+J^2)L^2+K^4-3J^2K^2+J^4)R+\\
& 4L^5+4K^5+4J^5-\\
& 4(L^4-3K^2L^2+K^4)\sqrt{L^2+K^2}-\\
& 4(L^4-3J^2L^2+J^4)\sqrt{L^2+J^2}-\\
& 4(K^4-3J^2K^2+J^4)\sqrt{K^2+J^2}-20J^3KL\pi\,.
\end{aligned}$$

with $R=\sqrt{J^2+K^2+L^2}$. For a unit cube A with size $1\times1\times1$ we write for the energy

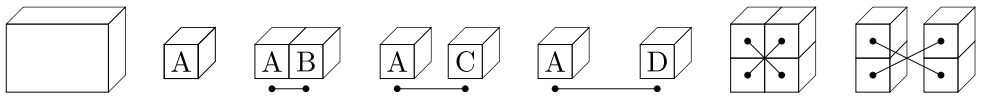
$$\bar{E}_{000}=\bar{E}_{\mathrm{AA}}$$

where E_{AA} denotes the self energy (see figure 2.38b). The total energy of two unit cubes with size $1\times1\times2$ consists of

$$\bar{E}_{001}=\underbrace{\bar{E}_{\mathrm{AA}}}_{\text{self A}}+\underbrace{\bar{E}_{\mathrm{BB}}}_{\text{self B}}+\underbrace{\bar{E}_{\mathrm{AB}}}_{\text{interaction AB}}+\underbrace{\bar{E}_{\mathrm{BA}}}_{\text{interaction BA}}\,.$$

Since $\bar{E}_{\mathrm{AA}}=\bar{E}_{\mathrm{BB}}$ and $\bar{E}_{\mathrm{AB}}=\bar{E}_{\mathrm{BA}}$ we obtain for the interaction energy of A and B (see figure 2.38c)

$$\bar{E}'_{001}=\bar{E}_{\mathrm{AB}}=\frac{1}{2}\left(\bar{E}_{001}-2\bar{E}_{\mathrm{AA}}\right)\,.$$



(a) $\bar{E}_{012}$ (b) $\bar{E}'_{000}$ (c) $\bar{E}'_{001}$ (d) $\bar{E}'_{002}$ (e) $\bar{E}'_{003}$ (f) $\bar{E}'_{011}$ (g) $\bar{E}'_{012}$

Figure 2.38.: Interaction sets for the (012) cube in (a). It consists of 6 self-interaction elements (b), 7 interaction elements of type (c), 2 interaction elements of type (d), none interaction element of type (e), 2 interaction elements of type (f) and 1 interaction element of type (g).

Now we are able to calculate another interaction energy $\bar{E}'_{002} = \bar{E}_{\mathrm{AC}}$ from $\bar{E}_{002}$ depicted in figure 2.38d with

$$\begin{aligned}\bar{E}_{002} &= \bar{E}_{\mathrm{AA}} + \bar{E}_{\mathrm{BB}} + \bar{E}_{\mathrm{CC}} + 2\bar{E}_{\mathrm{AB}} + 2\bar{E}_{\mathrm{BC}} + 2\bar{E}_{\mathrm{AC}} \\ &= 3\bar{E}_{\mathrm{AA}} \qquad\qquad + 4\bar{E}_{\mathrm{AB}} \qquad\qquad + 2\bar{E}_{\mathrm{AC}}\,.\end{aligned}$$

The interaction energy $\bar{E}'_{002} = \bar{E}_{\mathrm{AC}}$ can be computed directly since we have analytical expressions for $\bar{E}_{\mathrm{AA}}$, $\bar{E}_{\mathrm{AB}}$ and $\bar{E}_{002}$. The prefactors (3, 4, 2) can also be found in a reduced (symmetric) form (3, 2, 1) in table 2.5 in the first column.

The total energy of the set 012 can be divided in five subgroups. The subgroups are shown in figure 2.38 and in the second column of table 2.5. The energy $\bar{E}_{012}$ yields

$$\bar{E}_{012} = 6\bar{E}_{000} + 7\bar{E}'_{001} + 2\bar{E}'_{002} + 0\bar{E}'_{003} + 2\bar{E}'_{011} + 1\bar{E}'_{012}\,.$$

For the remaining interaction energies we can proceed in the same manner. However, we have to identify the number of interaction subsets for each newly calculated set. The number of interactions can be found in table 2.5. The interaction energies can be obtained with the help of table 2.5. We start by calculating the energies of $\bar{E}_{000}$ and $\bar{E}'_{001}$ as described above. Then, we compute additional interaction energies starting at the leftmost column until we end up at the last element $\bar{E}'_{333}$. All 16 interaction energies are now calculated and can be found in table 2.6.

Comparison of the Two Error Schemes

The result of the additional error scheme is plotted in figure 2.39. The introduced correction factor

$$\frac{(j+l)!}{j!\,l!\,2^{j+l+1}}$$

for the first error scheme is not valid for any order of multipoles considering only single box-box interactions. The error exceeds the computed limit at $p = 4$. However, assuming a homogeneous particle distribution in each box, changes the result slightly and validates the first error scheme. The result is plotted in figure 2.39b.

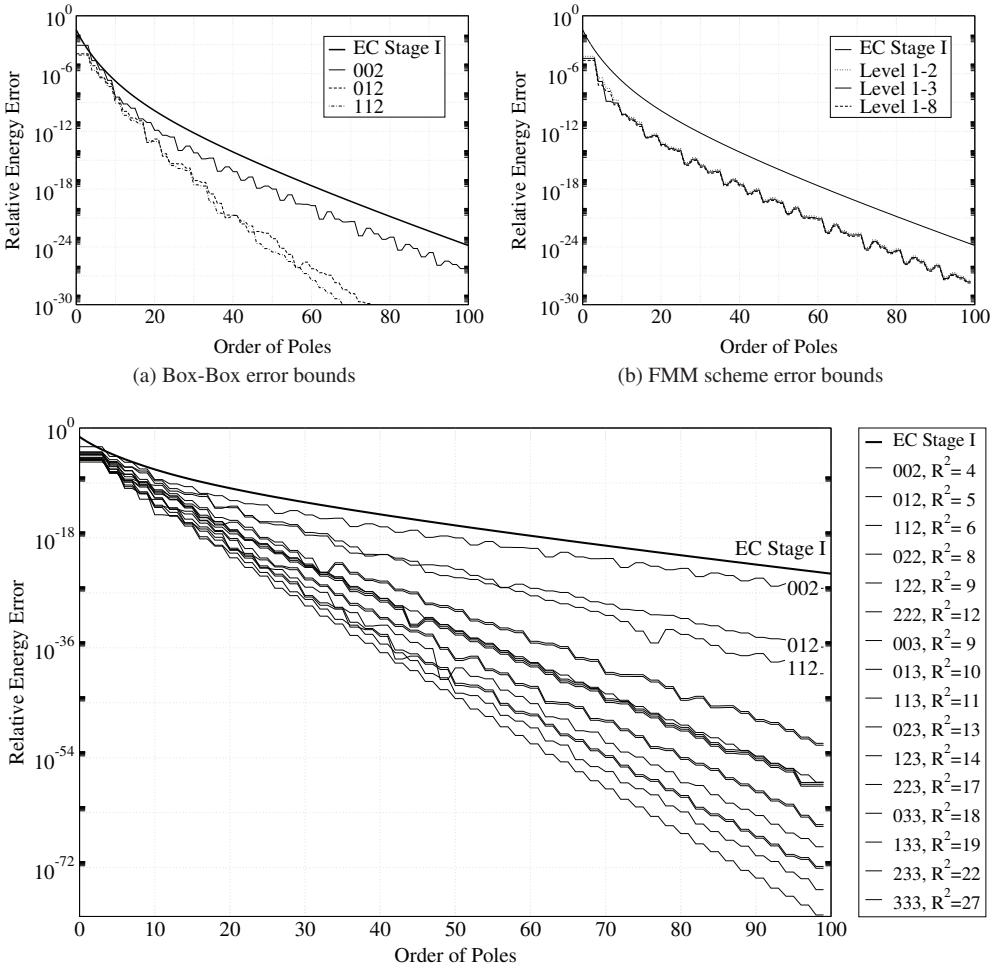
	Far Field Box Coordinates xyz															
xyz	002	003	012	013	022	023	033	112	113	122	123	133	222	223	233	333
000	3	4	6	8	9	12	16	12	16	18	24	32	27	36	48	64
001	2	3	7	10	12	17	24	20	28	33	46	64	54	75	104	144
002	1	2	2	4	6	10	16	4	8	12	20	32	27	42	64	96
003	–	1	–	2	–	3	8	–	4	–	6	16	–	9	24	48
011	–	–	2	3	4	6	9	11	16	20	29	42	36	52	75	108
012	–	–	1	2	4	7	12	4	8	14	24	40	36	58	92	144
013	–	–	–	1	–	2	6	–	4	–	7	20	–	12	34	72
022	–	–	–	–	1	2	4	–	–	2	4	8	9	16	28	48
023	–	–	–	–	–	1	4	–	–	–	2	8	–	6	20	48
033	–	–	–	–	–	–	1	–	–	–	–	2	–	–	3	12
111	–	–	–	–	–	–	–	2	3	4	6	9	8	12	18	27
112	–	–	–	–	–	–	–	1	2	4	7	12	12	20	33	54
113	–	–	–	–	–	–	–	–	1	–	2	6	–	4	12	27
122	–	–	–	–	–	–	–	–	–	1	2	4	6	11	20	36
123	–	–	–	–	–	–	–	–	–	–	1	4	–	4	14	36
133	–	–	–	–	–	–	–	–	–	–	–	1	–	–	2	9
222	–	–	–	–	–	–	–	–	–	–	–	–	1	2	4	8
223	–	–	–	–	–	–	–	–	–	–	–	–	–	1	4	12
233	–	–	–	–	–	–	–	–	–	–	–	–	–	–	1	6
333	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	1

Table 2.5.: Number of possible subbox positions (rows) $x_s \times y_s \times z_s$ in a cuboid box with dimension $x \times y \times z$ (columns). For $ws > 1$ additional box sizes have to be considered.

Additionally the scheme shows that for an increasing order of poles only the closest far field contributions (002) need to be taken into account. The total error follows the error for the (002) interaction for large multipoles precisely.

The derived error scheme presents the minimal order of poles, i.e. the lower bound, for uniformly distributed particle system. Since the simulation has to deal with a finite particle system the order of poles will be higher than the computed values.

In the next chapter we extend the FMM and stage I and II of the error-control scheme for periodic boundary conditions.



(a) Box-Box error bounds

(b) FMM scheme error bounds

(c) FMM scheme error bounds for each possible far field box separately

Figure 2.39.: Analytical error bound for homogenously distributed particle systems. Single box-box error bounds (a) do exceed the error bound given by stage I of the error-control scheme. However, taking into account the actual FMM interaction set, the result from stage I (b) does not exceed the lower limit provided by the analytical solution. For a high order of poles only the contributions from (002) box-box interactions dictate the error bound (c). The remaining contributions become insignificant with increasing order of poles.

xyz	R^2	$1/R$	Energy
000	0	∞	1.882 312 644 389 660 16
002	4	0.500 000 000 000 000 00	0.499 139 847 013 560 54
003	9	0.333 333 333 333 333 33	0.333 215 481 099 826 00
012	5	0.447 213 595 499 957 94	0.447 100 395 342 384 79
013	10	0.316 227 766 016 837 93	0.316 177 059 082 768 54
022	8	0.353 553 390 593 273 76	0.353 591 183 594 328 64
023	13	0.277 350 098 112 614 56	0.277 352 772 190 202 96
033	18	0.235 702 260 395 515 84	0.235 707 391 268 263 31
112	6	0.408 248 290 463 863 02	0.408 331 273 621 761 74
113	11	0.301 511 344 577 763 62	0.301 495 607 797 756 90
122	9	0.333 333 333 333 333 33	0.333 391 890 319 486 35
123	14	0.267 261 241 912 424 38	0.267 271 135 829 273 37
133	19	0.229 415 733 870 561 77	0.229 422 572 539 922 13
222	12	0.288 675 134 594 812 88	0.288 715 087 097 265 34
223	17	0.242 535 625 036 332 97	0.242 548 590 198 871 85
233	22	0.213 200 716 355 610 43	0.213 208 250 460 023 95
333	27	0.192 450 089 729 875 25	0.192 455 276 681 293 53

Table 2.6.: Box-Box Energy for two boxes with homogenously distributed charges and total charge $Q = \sum_{i=1}^{\infty} q = 1$. The third column characterizes the monopole-monopole interaction energy of two distinct boxes. Already three digits are valid compared to the analytical solution displayed in the last column. For $ws = 1$ all entries in the last column are sufficient to set up an error control scheme for the FMM. The contribution of each box-box interaction to the overall error is shown in figure 2.39c.

3. FMM and Periodic Boundary Conditions

The presented fast multipole scheme in chapter 2 for open boundary conditions can be extended to periodic boundary conditions (PBC) as well. The periodic boundaries allow to derive macroscopic bulk properties of the simulated particle system. A huge (even infinite) ensemble of particles is reduced to a smaller (finite) ensemble inside a finite-sized simulation box. This simulation box then is replicated in all spatial directions. Hence, the influence of the boundary on the particles enclosed in the finite simulation box vanishes and only the influence of the bulk properties remains. As a particle moves through a boundary into a neighboring image box, it will enter on the opposite site of the central simulation box, thus the number density is conserved.

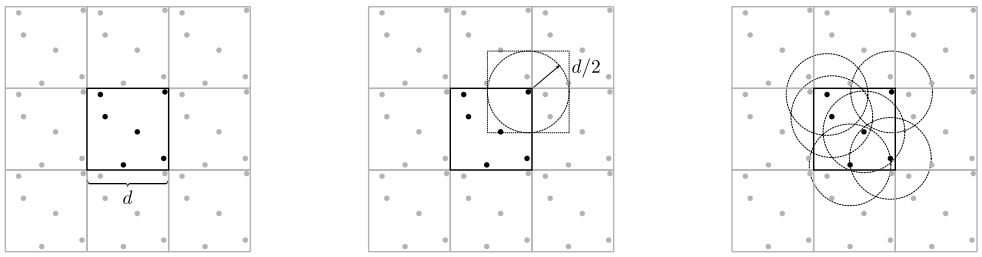
To fill the entire 3D space, several different shapes of the central simulation box may be considered [53]. However, we only derive the algorithm for the most common shape, i.e. the cubic box. The presented scheme may also be applied together with other shapes like parallelepipeds, hexagonal prisms, octahedrons or dodecahedrons.

Some simulations even demand a mixed boundary condition with periodic boundaries in only one or two dimensions, such as electrolyte solutions, membranes, nanopores or nanotubes [54]. The presented FMM scheme can also be applied to such mixed boundary systems. However, the derivation of the algorithm in this chapter is performed for three dimensional periodicity if not stated otherwise.

3.1. Minimum Image Convention

The direct computation of all mutual interactions of N particles in a system with open boundaries demands work proportional to N^2. The cut-off scheme presented in section 1.1.1 reduces the overall complexity to $\mathcal{O}(N)$. However, no *a priori* error bounds can be formulated.

The calculation of interactions in a periodic system via a direct method leads to new problems. Now, the number of particles involved is infinite. Therefore, we cannot compute the direct interactions straightforward, not even with the unfavorable $\mathcal{O}(N^2)$ complexity. However, we may apply the cut-off scheme together with a minimum image convention [55], which is displayed in figure 3.1. The energy of such a system is computed by taking only the closest particle neighbors into account. All interactions of pairs of particles with a distance larger than the spherical cut-off radius r_c are neglected, hence zero. The interaction region or image should have the same size and shape as the periodic simulation box. Therefore, the implied minimum image limits the cut-off to half the length d of the original simulation box. The overall complexity is reduced



(a) Simulation Box and Images (b) Minimum Image (c) Cut-off Interaction Set

Figure 3.1.: The minimum image convention only chooses the closest neighboring particles around each particle. The spherical cut-off radius r_c is limited to $r_c \leq d/2$ half the box size d. All pair potentials $\phi(r)$ with $r > r_c$ are neglected.

to $\mathcal{O}(N)$. For rapidly decaying short-range potentials, the scheme may generate fast and sufficiently precise results. However, long-range potentials will introduce large and uncontrollable errors to the results such as

- discontinuities in the potential energy,
- discontinuities in the forces,
- violation of energy conservation,
- and changes in the thermodynamic properties of the system.

Possible countermeasures exist, e.g. potential shift, but do not eliminate the inherent problem. For long-range potentials, the simulation box has to be expanded to minimize the listed errors, but for large systems such a scheme is impractical.

3.2. Definition of the Boundary Condition

The definition of the boundary conditions can be performed in a general manner. We only assume that the simulation cell must be translationally symmetric in order to fill the entire $\mathbb{R}^n$ space. No additional assumptions on the basis vectors forming the periodic lattice are necessary.

3.2.1. Three Dimensional Periodicity

We define a simulation cell $\Gamma(\mathbf{0})$ in three dimensions and the basis vectors $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ with

$$\Gamma(\mathbf{0}) = \{\mathbf{r} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + x_3\mathbf{a}_3 : \, -1/2 \leq x_i \leq 1/2, \, \text{for}\, i = 1, 2, 3\}\,.$$

Additionally, we chose $\mathbf{a}_i$ in such a way that the volume V_Λ of the simulation cell is defined by $V_\Lambda = \mathbf{a}_1 \cdot (\mathbf{a}_2 \times \mathbf{a}_3) > 0$. Now, we set up a lattice Λ of translationally symmetric copies $\Gamma(\mathbf{n})$

with $\mathbf{n} = n_1\mathbf{a}_1 + n_2\mathbf{a}_2 + n_3\mathbf{a}_3$ and $n_i \in \mathbb{Z}$ of the original central simulation cell $\Gamma(\mathbf{0})$ by

$$\Gamma(\mathbf{n}) = \{\mathbf{r} : \mathbf{r} - \mathbf{n} \in \mathbf{\Gamma}(\mathbf{0})\}\,.$$

Each replica cell $\Gamma(\mathbf{n})$ contains the exact same numbers of particles as the simulation cell $\Gamma(\mathbf{0})$. A particle with position $\mathbf{r}_i$ in the central simulation cell $\Gamma(\mathbf{0})$ has a replica particle at $\mathbf{r}_i + \mathbf{n}$ in the lattice cell $\Gamma(\mathbf{n})$.

3.2.2. Two Dimensional Periodicity

For the periodicity in two dimensions, we follow the derivation of the three dimensional case. Without loss of generality, we establish the periodicity in the xy-plane and define the central simulation cell $\Gamma(\mathbf{0})$ again as

$$\Gamma(\mathbf{0}) = \{\mathbf{r} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + x_3\mathbf{a}_3 :\ -1/2 \leq x_i \leq 1/2,\ \text{for}\, i = 1, 2, 3\}\,.$$

The corresponding lattice vectors in the xy-plane are $\mathbf{n} = n_1\mathbf{a}_1 + n_2\mathbf{a}_2$ with $n_i \in \mathbb{Z}$. The periodic lattice cells $\Gamma(\mathbf{n})$ are given by

$$\Gamma(\mathbf{n}) = \{\mathbf{r} : \mathbf{r} - \mathbf{n} \in \mathbf{\Gamma}(\mathbf{0})\}\,.$$

3.2.3. One Dimensional Periodicity

Without loss of generality, we choose the periodicity axis along the z-axis in a one dimensional periodic system and define the central simulation cell $\Gamma(0)$ as

$$\Gamma(\mathbf{0}) = \{\mathbf{r} = x_1\mathbf{a}_1 + x_2\mathbf{a}_2 + x_3\mathbf{a}_3 :\ -1/2 \leq x_i \leq 1/2,\ \text{for}\, i = 1, 2, 3\}\,.$$

The periodic images (replica cells) of the central cell $\Gamma(\mathbf{0})$ are given by

$$\Gamma(\mathbf{n}) = \{\mathbf{r} : \mathbf{r} - \mathbf{n} \in \Gamma(\mathbf{0})\}\,,$$

with $\mathbf{n} = n_3\mathbf{a}_3$. Since we want to derive our algorithm for the cubic box only, all basis vectors are perpendicular to one another and our simulation box has a unit volume $V_\Lambda = 1$.

3.2.4. Periodic Potential and Energy

We now define the potential $\Phi(\mathbf{R})$ inside a lattice box $\Gamma(\mathbf{n})$ for a periodic system as

$$\Phi(\mathbf{R}) = \sum_{\mathbf{n}} \sum_{j=1}^{N} \frac{q_j}{|\mathbf{R} - \mathbf{r}_j + \mathbf{n}|}$$

as well as the energy

$$E = \frac{1}{2} \sum_{\mathbf{n}} \sum_{i=1}^{N} {\sum_{j=1}^{N}}' \frac{q_i q_j}{|\mathbf{r}_i - \mathbf{r}_j + \mathbf{n}|}$$

where the prime indicates that for the central box with $\mathbf{n} = \mathbf{0}$ the singularity for $i = j$ is dropped.

3.3. Convergence of Lattice Sums

Since the potential $\Phi(\mathbf{R})$ does not satisfy

$$|\Phi(\mathbf{R})| \leq A\,|\mathbf{r}|^{-d-\epsilon} \tag{3.1}$$

with d being the periodicity and $A, \epsilon > 0$ for an arbitrary set of point particles with non-zero monopole, dipole- or quadrupole moments, we have to add additional constraints. The following interactions do not obey the inequality in (3.1):

- Charge–Charge Interactions (1D, 2D, 3D) $|\mathbf{r}|^{-1}$
- Charge–Dipole Interactions (2D, 3D) $|\mathbf{r}|^{-2}$
- Dipole–Dipole Interactions (3D) $|\mathbf{r}|^{-3}$
- Charge–Quadrupole Interactions (3D) $|\mathbf{r}|^{-3}$

Therefore, we add the additional constraint for all particles in the simulation cell

$$Q = \sum_{i=1}^{N} q_i = 0\,.$$

Hence, with box net charge $Q = 0$ only the dipole terms do not converge absolutely. We neglect the dipole-dipole contributions temporarily, but will get back to them later.

Speed of Convergence

Since we want to simulate periodic systems under the influence of long-range forces even for machine precision, we have to include more and more lattice cells $\Gamma(\mathbf{n})$ in a direct summation to obtain a sufficient precision. For a Madelung system [56] an upper bound for the speed of convergence can be found. We can define a simple periodic lattice by a three-dimensional Epstein zeta function

$$M(s) = \sum_{i,j,k\in\mathbb{Z}}{}' \frac{(-1)^{i+j+k}}{(i^2+j^2+k^2)^s}\,.$$

The prime again indicates that the singularity at the origin is to be avoided. The special case $M(1/2)$ is called Madelung constant. $M(1/2)$ represents the potential energy at the origin in a lattice of alternating unit charges. Clearly $M(s)$ converges absolutely for all $s > 3/2$, but does not converge for ever-expanding spheres. Borwein et. al. [56] showed that the sum converges for ever-expanding cubes. However, no closed form exists for the 3D case. Let us for the moment assume we want to tackle the problem by a direct approach, hence adding more and more lattice cells (cubes). Since we carry out all operations on a limited precision machine, we may ask how many layers of cubes are necessary to obtain the requested precision. Borwein et. al. estimated the number of cubes as

$$\left| M(1/2) - \sum_{|i|,|j|,|k|<m}{}' \frac{(-1)^{i+j+k}}{(i^2+j^2+k^2)^{1/2}} \right| < \frac{24}{m}$$

with $m_a > 0$ for $a = \{1, 2, 3\}$ and $m = \min\{m_1, m_2, m_3\}$. Since the convergence only depends on the minimal expanded dimension m, we are also able to utilize shapes other than cubes, e.g. rectilinear shapes. However, the slow convergence demands different approaches to solve the problem in acceptable time.

3.4. Ewald-Based Summation Schemes

The most famous scheme to approach the problem is the Ewald summation technique [57]. It allows to reduce the complexity to $\mathcal{O}(N \log N)$. The main idea behind the scheme is to convert the sum into two absolutely convergent series. One part is evaluated in real space the other part is evaluated in reciprocal space. The conversion is performed by representing all occurring point charges as a Gaussian charge density decaying exponentially. A second Gaussian charge density is applied to cancel the newly introduced function after the computation. The influence of the canceling contribution is computed in reciprocal space, since it represents a smooth periodic function which can be transformed into a fast converging Fourier series. This identity can be written as

$$\Phi(r) = \frac{1}{r} = \frac{f(r)}{r} + \frac{1 - f(r)}{r} = \Phi_{\text{recip}}(r) + \Phi_{\text{real}}(r)\,.$$

The function $f(r)$ used to split the potential is the error function $\operatorname{erf}(\beta r)$. The introduced parameter β is needed to control the precision of the result. We obtain for the real-space potential

$$\Phi_{\text{real}}(r) = \frac{1}{r} - \frac{\operatorname{erf}(\beta r)}{r} = \frac{\operatorname{erfc}(\beta r)}{r}$$

and the reciprocal space potential

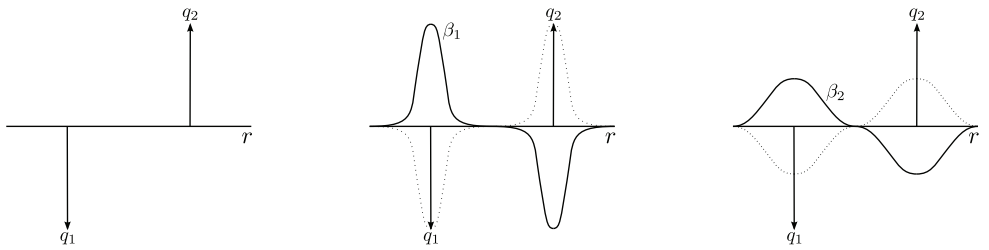
$$\Phi_{\text{recip}}(r) = \frac{\operatorname{erf}(\beta r)}{r}\,.$$

However, the scheme does not fit our requirements. The desired complexity should be of order N instead of $N \log N$. Besides, the given formulation does not allow for an FMM summation scheme for the real-space part. Yet, the classical Ewald scheme can be adapted to the FMM in three dimensions. In doing so, we must introduce a lattice operator for the periodic contributions. Such a scheme is dicussed in the following section.

FMM + Ewald-like Summation

To formulate a suitable set of equations we recall the definition of the periodic potential

$$\Phi(\mathbf{R}) = \sum_{\mathbf{n}} \sum_{j=1}^{N} \frac{q_j}{|\mathbf{R} - \mathbf{r}_j + \mathbf{n}|}\,.$$



(a) particle system (b) Masked Gaussians with β_1 (c) Masked Gaussians with β_2

Figure 3.2.: The conditionally convergent sums are converted into two rapidly and absolutely convergent terms. Each point charge (a) is masked by a Gaussian function (b) and (c). Depending on the splitting parameter β, the width of charge distribution changes. The solid lines in (b) and (c) represent contributions calculated in real space. The dotted lines represent contributions calculated in reciprocal space.

Additionally, we do know the bipolar expansion for an open particle system (see figure 3.3) with

$$\begin{aligned}\frac{1}{|\mathbf{r}' - \mathbf{a}_1|} &= \frac{1}{|\mathbf{R} + \mathbf{a}_2 - \mathbf{a}_1|} \\ &= \sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\sum_{m=-l}^{l}\sum_{k=-j}^{j} (-1)^j \omega_{lm}(\mathbf{a}_1)\mathcal{B}_{l+j,m+k}(\mathbf{R})\omega_{jk}(\mathbf{a}_2)\,.\end{aligned}$$

Since we can substitute $\mathbf{R}$ with $\mathbf{n}$ for the $\mathcal{B}$ operator, we are able to define a lattice operator $\mathcal{L}[\mathbf{n}]$ including all contributions from the entire lattice. The bipolar expansion of the lattice cells reads

$$\frac{1}{|\mathbf{a}_2 - \mathbf{a}_1 + \mathbf{n}|} = \sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\sum_{m=-l}^{l}\sum_{k=-j}^{j} (-1)^j \omega_{lm}(\mathbf{a}_1)\mathcal{L}_{l+j,m+k}(\mathbf{n})\omega_{jk}(\mathbf{a}_2)\,. \tag{3.2}$$

At this point we do not need to keep the indices of the multipole moments $\omega_{lm}(\mathbf{a}_1)$ and $\omega_{lm}(\mathbf{a}_2)$. Both expansions originate from the same particles, one representing the simulation box and the other one representing a periodic image. Thus, we can just use one common expansion $\omega_{lm}(\mathbf{a})$. Since the sums from the last equation are absolutely convergent, if

$$|\mathbf{a}_1| + |\mathbf{a}_2| = 2|\mathbf{a}| < |\mathbf{n}|\,,$$

we are able to setup a partition for the periodic lattice satisfying this condition. The partition is shown in figure 3.4 and divides the entire $\mathbb{R}^n$ space into two regions. Interactions in a lattice region with $\mathbf{n} = \mathbf{0}$ can be computed with the original FMM approach for open boundaries E_{open}. Contributions in the first layer E_{layer1} of the lattice with $\max(n_1, n_2, n_3) = 1$ can be computed with the original approach. However, only half of the energy belongs to the lattice energy of the

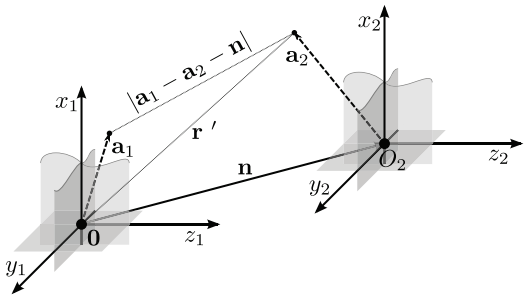


Figure 3.3.: Equation (3.2) is only valid, if we use the corresponding coordinate systems for the multipole expansions $\omega_{lm}(\mathbf{a}_1)$ and $\omega_{lm}(\mathbf{a}_2)$. The equation demands expansion one to be in the coordinate system O_1 with center $\mathbf{0}$ and expansion two in coordinate system O_2.

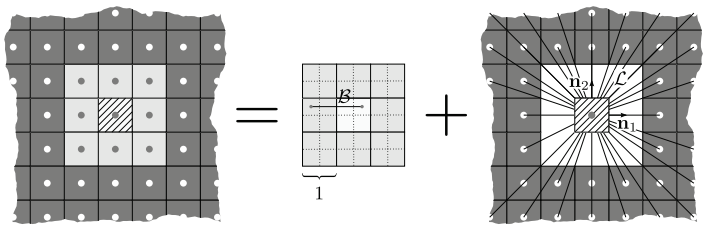


Figure 3.4.: (left) The simulation box and its replicas fill the entire space. The space is divided in two regions, a near field region (light grey) and a lattice region (dark grey).

central box. The remaining interactions from lattice cells E_{lattice} with $\max(n_1, n_2, n_3) > 1$ can be computed via the introduced $\mathcal{L}$ operator. The partitioned energy can be expressed as

$$E = E_{\text{open}} + E_{\text{layer1}} + E_{\text{lattice}}$$

with

$$E_{\text{lattice}} = \frac{1}{2} \sum_{l=0}^{\infty} \sum_{j=0}^{\infty} \sum_{m=-l}^{l} \sum_{k=-j}^{j} (-1)^j \omega_{lm}(\mathbf{a}) \mathcal{L}_{l+j,m+k}(\mathbf{n}) \omega_{jk}(\mathbf{a}) \,.$$

The operator $\mathcal{L}$ is a universal constant of the cell geometry. It can be precomputed for each lattice type up to a certain truncation order p by a number of approaches. Since our main goal – the implementation of an FMM implementation for periodic boundaries with error control – demands a parameter free approach for the lattice contributions, most of the available approaches have to be dropped out. Introducing the additional demand that even mixed periodicities in one- or two dimensions should be handled as easily as the 3D case, narrows the available approaches further. Ewald-based schemes do not provide advantageous algorithms for one- or two-dimensional periodicities [58], especially when combined with a FMM scheme.

3.5. Parameter-Free Renormalization Approach

We follow the approach proposed by Kudin and Scuseria [59] and incorporate the algorithm later on into our error control scheme.

3.5.1. Informal Description

The idea for the evaluation of lattice sums with a renormalization approach was first proposed by Berman and Greengard [60]. The scheme reduces the infinite summation of lattice sites to a rapidly converging finite summation yielding the lattice operator $\mathcal{L}$. The potential $\Phi(\mathbf{0})$ at the center of the central box $(0, 0, 0)$ can be computed by adding up contributions from lattice supercells of size $(2ws + 1)^j \times (2ws + 1)^j \times (2ws + 1)^j$ with $j \in \mathbb{N}$ going to infinity. Since we want to translate the multipoles from lattice cells into Taylor-like (local) expansions around the common center $(0, 0, 0)$ of the central box, we can add up all contributions into a single translation operator $\mathcal{L}$. We do not add up an infinite number of lattice sites, however, the fast convergence allows to precompute the lattice operator in machine precision. Since the available number of digits is fixed, we can call a result below machine precision numerically exact and therefore no additional runtime parameter is introduced.

3.5.2. Mathematical Operators

Since this approach is based on the original multipole scheme, we can reuse all introduced FMM translation and conversion operators from chapter 2. The FMM operators are valid on multipole

moments ω_{lm} and Taylor-like expansions μ_{lm} and their chargeless counterparts O_{lm} and M_{lm}. For simplicity we only use the chargeless multipole moments O_{lm} and chargeless Taylor-like coefficients M_{lm} in this section.

Translation of a Multipole Expansion

Let us recall the translation of a multipole expansion $O_{jk}(\mathbf{a})$ at $\mathbf{a}$ to a multipole expansion around a new center $\mathbf{a}+\mathbf{b}$

$$O_{lm}(\mathbf{a}+\mathbf{b}) = \sum_{j=0}^{l} \sum_{k=-j}^{j} O_{jk}(\mathbf{a}) O_{l-j,m-k}(\mathbf{b}) \,.$$

We are dropping the indices and abbreviate the operator for the following lattice sum algorithm via $\triangleleft$ into the form

$$O(\mathbf{a}+\mathbf{b}) = \mathcal{A}(\mathbf{b}) \triangleleft O(\mathbf{a}) \,.$$

Conversion of a Multipole Expansion into a Local Expansion

A multipole expansion at $\mathbf{a}$ transforms into a Taylor-like local expansion at $\mathbf{b}-\mathbf{a}$ via

$$M_{lm}(\mathbf{b}-\mathbf{a}) = \sum_{j=0}^{\infty} \sum_{k=-j}^{j} M_{j+l,k+m}(\mathbf{b}) O_{jk}(\mathbf{a}) \,.$$

Again, we abbreviate the operator into the form

$$M(\mathbf{b}-\mathbf{a}) = \mathcal{B}(\mathbf{b}) \otimes O(\mathbf{a}) \,.$$

Rescale a Multipole Expansion

Since we need a hierarchy of boxes and do not want to recompute multipole moments for each hierarchy level we introduce a scaling operator for a multipole expansion with

$$\mathcal{S}_O(O_{lm}(\mathbf{a})) = 3^l \cdot O_{lm}(\mathbf{a}) \,.$$

Rescale a Local Expansion

For similar reasons, the rescaling has to be performed for the local expansion M_{lm} as well. Therefore, we introduce a scaling operator for a Taylor-like expansion by

$$\mathcal{S}_L(M_{lm}(\mathbf{b})) = \frac{M_{lm}(\mathbf{b})}{3^{l+1}} \,.$$

We limit ourselves to the case where $ws = 1$. For a larger separation criterion ws, we have to substitute 3^l and 3^{l+1} with $(2ws+1)^l$ and $(2ws+1)^{l+1}$ accordingly.

Operator Properties

The operators derived in chapter 2 and the additionally introduced operators $\mathcal{S}_L$ and $\mathcal{S}_O$ are linear operators. Therefore, the following properties apply for infinite expansions [59]

$$M \otimes [O_1 \triangleleft O_2] = [M \otimes O_1] \otimes O_2 \tag{3.3}$$
$$O_1 \triangleleft [O_2 \triangleleft O_3] = [O_1 \triangleleft O_2] \triangleleft O_3 \tag{3.4}$$
$$\mathcal{S}_L(M) \otimes \mathcal{S}_O(O) = \mathcal{S}_L(M \otimes O) \tag{3.5}$$

where M represents any Taylor-like local expansion and O represents a certain multipole expansion which still gives a convergent expansion when shifted or transformed. The hierarchy of lattice boxes introduced in the next section guarantees that the speed of convergence and precision is not degraded by the use of the operator properties.

3.5.3. Lattice Sum Algorithm

Let us derive the local moment in the central cell $\Gamma(\mathbf{0})$ at the center $(0,0,0)$ due to an arbitrary lattice cell $\Gamma(\mathbf{n})$ with $\mathbf{n} \neq \mathbf{0}$ and $\mathbf{n} = j\mathbf{a}_1 + k\mathbf{a}_2 + l\mathbf{a}_3,\ j,k,l \in \mathbb{Z}$. We define the contribution with the help of the $\mathcal{B}$ operator as

$$L^{(j,k,l)} = \mathcal{B}(\mathbf{n}) \otimes \omega$$

with ω being the total multipole moment of the lattice cell. Since all boxes are images of the original simulation box at the center, the multipole expansion of each image box is given by

$$\omega(\mathbf{n}) \equiv \omega(\mathbf{0})\,.$$

The contribution from all lattice cells for 1D-, 2D- and 3D-periodic systems reads

$$\begin{aligned}
\text{1D}: \quad L^{\infty} &= \sum_{|\max(0,0,l)|>1} \mathcal{B}((0,0,l)) \otimes \omega(\mathbf{a}) \\
\text{2D}: \quad L^{\infty} &= \sum_{|\max(j,k,0)|>1} \mathcal{B}((j,k,0)) \otimes \omega(\mathbf{a}) \\
\text{3D}: \quad L^{\infty} &= \sum_{|\max(j,k,l)|>1} \mathcal{B}((j,k,l)) \otimes \omega(\mathbf{a})\,.
\end{aligned}$$

All nearest neighbors for the central box, i.e. the first layer are excluded to guarantee convergence of the expansions. Since the $\mathcal{B}$ operator is a linear operator, it allows us to precalculate $\mathcal{B}$ for the entire lattice before applying the unique multipole expansion ω. However, applying a direct space summation like in the last set of equations is difficult, because the convergence of the lattice sum is very slow. However, the evaluation of the lattice sum can be rapidly achieved by introducing a hierarchy of supercells with variable size $(2ws+1)^j \times (2ws+1)^j \times (2ws+1)^j$.

Since we are only interested for the case where $ws = 1$, we use a hierarchy of size $3^j \times 3^j \times 3^j$. We define an interaction set as follows. For the first interaction set Λ_0 we include all lattice cells in the far field of the central (0,0,0) cell but ignore all cells in the near field of the supercell $3^1 \times 3^1 \times 3^1$. This rule is equivalent to the partitioning rule in chapter 2 for open boundaries. We write

$$L_* = \sum_{\Lambda_0} \mathcal{B}((j,k,l)) \,.$$

The far field contribution from region Λ_0 is then

$$L^0 = L_* \otimes \omega \,.$$

We proceed on the next supercell level for $j = 1$ with cell size $3^1 \times 3^1 \times 3^1$. Again we only add cells which are in the near field of the $3^2 \times 3^2 \times 3^2$ supercell. The total contribution to the lattice operator L after this step reads

$$L^1 = L_* \otimes \omega + \mathcal{S}_L(L_*) \otimes [O_* \triangleleft \omega] \,.$$

The second contribution demands scaled local moments L_*. At the same time we combine multipole moments from a $3^0 \times 3^0 \times 3^0$ supercell into multipoles of a $3^1 \times 3^1 \times 3^1$ supercell. O_* is defined for each periodic boundary condition by

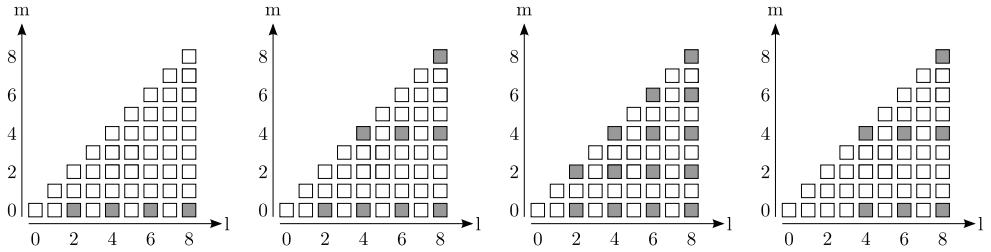
$$\begin{aligned}
\text{1D}: \quad O_* &= \sum_{\Omega_1 = -1 \le l \le 1} \mathcal{A}((0,0,l)) \\
\text{2D}: \quad O_* &= \sum_{\Omega_1 = -1 \le j,k \le 1} \mathcal{A}((j,k,0)) \\
\text{3D}: \quad O_* &= \sum_{\Omega_1 = -1 \le j,k,l \le 1} \mathcal{A}((j,k,l)) \,.
\end{aligned}$$

Now, we move on to the next level of supercells given by

$$\begin{aligned}
L^2 = & L_* \otimes \omega + \\
& \mathcal{S}_L(L_*) \otimes [O_* \triangleleft \omega] + \\
& \mathcal{S}_L(\mathcal{S}_L(L_*)) \otimes [\mathcal{S}_O(O_*) \triangleleft [O_* \triangleleft \omega]] \,.
\end{aligned}$$

Applying the operator properties from (3.5) yields expressions of the form $\mathcal{L}^n = L^n \otimes \omega$. We identify the following partial sums

$$\begin{aligned}
\mathcal{L}^0 &= L_* \\
\mathcal{L}^1 &= L_* + \mathcal{S}_L(L_*) \otimes O_* \\
\mathcal{L}^2 &= L_* + \mathcal{S}_L(L_*) \otimes O_* + \mathcal{S}_L(\mathcal{S}_L(L_*)) \otimes [\mathcal{S}_O(O_*) \triangleleft O_*] \\
&= L_* + \mathcal{S}_L(L_* + \mathcal{S}_L(L_*) \otimes O_*) \otimes O_* \\
&= L_* + \mathcal{S}_L(\mathcal{L}_1) \otimes O_* \,.
\end{aligned}$$



(a) $\mathcal{L}_{lm}$ 1D, z-axis (b) $\mathcal{L}_{lm}$ 2D, xy-axes (c) $\mathcal{L}_{lm}$ 2D, yz-axes (d) $\mathcal{L}_{lm}$ 3D, xyz-axes

Figure 3.5.: Depending on the periodicity and the periodic axes the lattice operator $\mathcal{L}_{lm}$ has only few non-zero entries. These elements are shown in the gray boxes. Empty boxes represent zero entries.

Again, the operator properties allow us to establish an infinite recursion scheme to set up the lattice sum $\mathcal{L}^n$ for increasing n with

$$\begin{aligned} \mathcal{L}^0 &= L_* \\ \mathcal{L}^{n+1} &= \mathcal{S}_L(\mathcal{L}^n) \otimes O_* + L_* \,. \end{aligned}$$

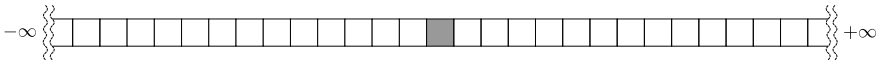
The full infinite lattice is defined via the lattice operator

$$\mathcal{L} = \mathcal{L}^\infty \,.$$

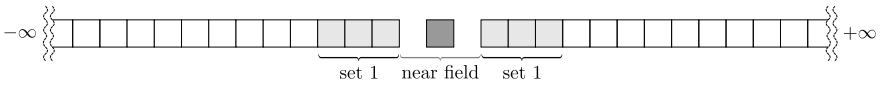
Since we want to use the result on a limited precision machine, we halt the recursion after a certain precision ε has been reached. The convergences of the lattice sum $\mathcal{L}$ increases with increasing multipole order. Therefore, we have to perform the precision check for the low-order elements. Since, by definition, the monopole element $\mathcal{L}_{00}$ does not converge and due to symmetry not all elements $\mathcal{L}_{lm}$ are non-zero, we halt the recursive scheme when

$$\begin{aligned} \text{1D}: &\quad |\mathcal{L}^{n+1}_{2,0} - \mathcal{L}^n_{2,0}| < \varepsilon \\ \text{2D}: &\quad |\mathcal{L}^{n+1}_{4,0} - \mathcal{L}^n_{4,0}| < \varepsilon \\ \text{3D}: &\quad |\mathcal{L}^{n+1}_{4,0} - \mathcal{L}^n_{4,0}| < \varepsilon \,. \end{aligned}$$

The precision goal we used to obtain the unique lattice operator was set to $\varepsilon = 10^{-64}$. The results of the calculation can be found in the appendix. Due to symmetry considerations many elements of $\mathcal{L}$ are zero. The non-zero elements are shown in figure 3.5d. We selected the periodicity axes in such a way that the number of non-zero items is minimal.

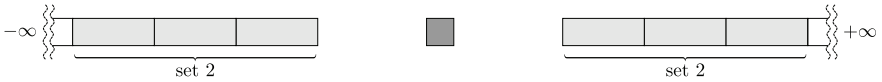


(a) Expand simulation box (gray) to infinity in all periodic dimensions.

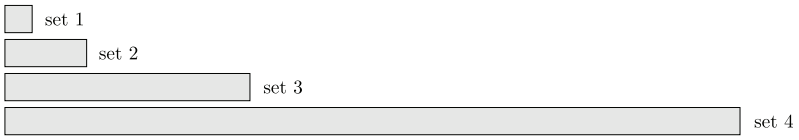


(b) Define first interaction set (light gray) and drop near field

(c) Combine multipole expansions

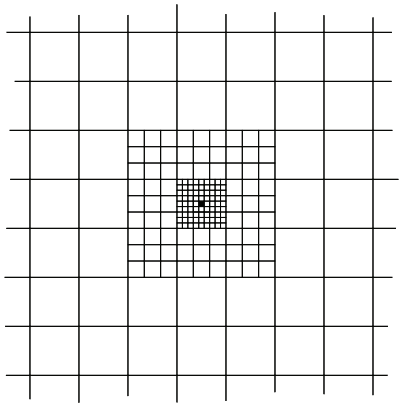


(d) Define second interaction set (light gray)

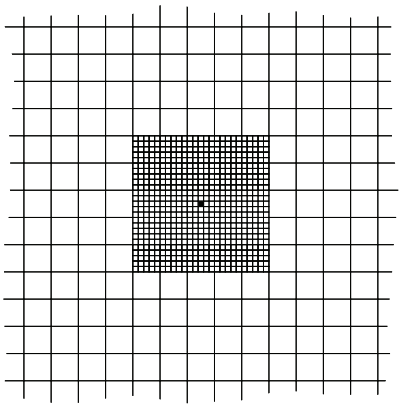


(e) Size of interaction sets

Figure 3.6.: 1D representation for the computation of the lattice sum. Figure (a) shows the expansion of the (grayed) central simulation box to infinity. Figure (b) shows the dropped nearest neighbors of the simulation box and the first interaction set. The second interaction set (c)–(d) is generated by combining multipole expansions. Figure (e) shows the increasing size of the interaction sets used with increasing distance to the central simulation cell.



(a) Periodic subsets $j = 1, 2$ and partly 3 for separation criterion $ws = 1$

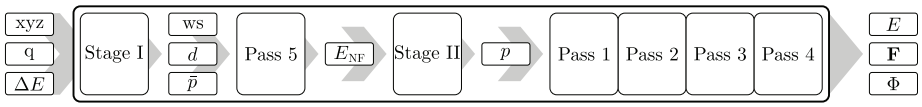


(b) Periodic subsets $j = 1$ and partly 2 for separation criterion $ws = 2$

Figure 3.7.: Interaction set for (a) $ws = 1$ with size $3^j \times 3^j \times 3^j$ and (b) $ws = 2$ with size $5^j \times 5^j \times 5^j$.

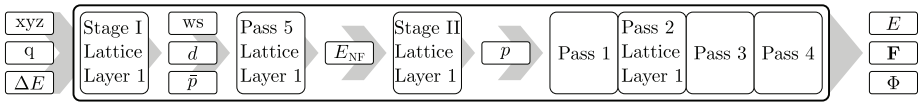
3.6. Implementation Details

To establish an FMM with periodic boudary conditions we have to add or modify certain parts of the implementation in section 2.6. The error-controlled FMM for open boundaries can be described by the following flow chart:



The periodic FMM needs changes to the scheme. In addition to the original approach we have to add contributions from the lattice and the first layer around the central box. The changes occur in

- the first stage of the error control,
- pass 5,
- the second stage of the error control and
- pass 2.



Pass 1 has to be modified slightly to shift the multipole expansions up the tree until depth $d = 0$ is reached. In a related manner, pass 3 has to collect Taylor-like expansions starting from tree depth $d = 0$.

3.6.1. Additional FMM Pass for the Lattice Operator

Our current implementation uses precomputed values for the lattice operator $\mathcal{L}$ in pass 2. Since the implementation is based on cubic boxes, currently no other cell shapes are available. The extension to different lattice cell shapes demands extra effort for the lattice operator $\mathcal{L}$. It is possible to precompute $\mathcal{L}$ for all required lattices or compute $\mathcal{L}$ directly in the simulation. Therefore, an additional pass has to be added prior to the error control to compute the missing $\mathcal{L}$ terms. The workload of the additional pass is independent of the number of particles. It has a complexity of $\mathcal{O}(p^4)$ with respect to the order of poles p.

3.6.2. Modifications of FMM Pass 1–5

Pass 1

All multipole expansions have to be shifted to the highest level $d = 0$. The total charge of the simulation cell (ω_{00}) has to be zero in order to guarantee convergence.

Pass 2

In addition to the interactions inside the simulation box, far field contributions from image boxes have to taken into account. These interactions will occur on all tree levels. Furthermore, interactions from the lattice with the multipole expansion of the simulation box have to calculated.

Pass 3

The computed Taylor-like expansions in pass 2 have to be shifted into the lowest level boxes. Since these expansions were generated on all levels, pass 3 translates expansions starting at depth $d = 0$.

Pass 4

No changes occur in pass 4. All computed and shifted local moments are combined with the multipole moments to yield the far field energy, forces and potentials of the system

Pass 5

Similarly to pass 2, the near field computation has to take into account interactions from replica particles in image boxes.

3.6.3. Fractional Tree Depth and Periodic Boundary Conditions

The FMM for open systems allows to use fractional instead of integral tree levels by shrinking the full set of particles accordingly. However, the periodic FMM does not allow such a feature, since a shrinking operation moves particles from image boxes into the simulation box. Since we cannot create empty boxes by shrinking, the tree depth will take only integral values.

3.6.4. Dipole Correction

Until now we neglected the conditionally convergent dipole-dipole interactions. However, since we added the lattice in a spherical manner the element $\mathcal{L}_{20}$ vanishes. A slabwise summation would yield a different result with $\mathcal{L}_{20} \neq 0$. Unfortunately, the results of such a periodic FMM computation are still not comparable with a standard Ewald summation scheme at this stage [61]. To compare results we have to transform the obtained extrinsic energy into an intrinsic energy obtained by an Ewald summation via

$$E_{\text{in}} = E_{\text{ex}} - \frac{2\pi}{3}\mathbf{d}\cdot\mathbf{d}$$

with $\mathbf{d} = \sum_i q_i \mathbf{a}_i$ being the dipole moment of the simulation cell. For the gradient at particle position $\mathbf{x}_i$ the correction is given by

$$\nabla_i E_{\text{in}} = \nabla_i E_{\text{ex}} - \frac{4\pi}{3} q_i \mathbf{d}\,. \tag{3.6}$$

The same correction has to be applied for the potential

$$\Phi_{\text{in}}(\mathbf{R}) = \Phi_{\text{ex}}(\mathbf{R}) - \frac{4\pi}{3}(\mathbf{R}-\mathbf{R}_0)\cdot\mathbf{d} + \frac{2\pi}{3}Q$$

with $Q = \sum_i q_i \mathbf{a}_i \cdot \mathbf{a}_i$ being the trace of the Cartesian quadrupole tensor and $\mathbf{R}_0$ being the origin of the coordinate system.

3.7. Periodic Boundaries and Error Control

The derived lattice operator allows for a straightforward implementation of the error control scheme developed for open boundaries. The two-stage model remains intact, the derivation can be performed almost identical.

3.7.1. Stage I of the Error Estimation Scheme

We rewrite the expressions for a bipolar box-box expansion as

$$E_{\text{lattice}} = \frac{1}{2}\sum_{l=0}^{\infty}\sum_{m=-l}^{l}\sum_{j=0}^{\infty}\sum_{k=-j}^{j}(-1)^j\,\omega_{lm}(\mathbf{a}_1)\,\mathcal{L}_{l+j,m+k}\,\omega_{jk}(\mathbf{a}_2)\,.$$

Now, box 1 represents the simulation box and box 2 represents a box in the lattice. The lattice operator converts each contribution from all equal lattice boxes, i.e. box 2, into a single local expansion around the center of the simulation box. We define averaged multipole moments with

$$\bar{\omega}_{lm} = \left(\sum_{i=1}^{N} |q_i| \right) \int_0^1 \int_0^1 \int_0^1 \left(\frac{a}{2} \right)^l \left| \tilde{P}_{lm}(\cos \alpha) \right| \, \mathrm{d}x \, \mathrm{d}y \, \mathrm{d}z \, .$$

Again, we add a correction term to compensate for the overestimation of the averaged multipole moments

$$\frac{(j+l)!}{j!\, l!\, 2^{j+l+1}} \, .$$

In contrast to the error scheme with open boundaries we do not need derive a special $\mathcal{B}$ operator, since all contributions are already convoluted inside the lattice operator $\mathcal{L}_{lm}$. The averaged chargeless version of the multipole expansions allows us to derive a corresponding lattice energy as follows

$$\bar{E}_{\text{lattice}} = \frac{1}{2} \sum_{l=0}^{\infty} \sum_{m=-l}^{l} \sum_{j=0}^{\infty} \sum_{k=-j}^{j} \frac{(j+l)!}{j!\, l!\, 2^{j+l+1}} \bar{O}_{lm} | \mathcal{L}_{l+j,m+k} | \bar{O}_{jk} \, .$$

In contrast to open boundary conditions, we do not need to introduce the element g_{00} for the first term with $l = j = 0$ since

$$g_{00} = |\mathcal{L}_{00}| = 0 \, .$$

However, the remaining elements g_{jl} for the inner sums in the second term are necessary and are defined by

$$g_{jl} = \sum_{m=-l}^{l} \sum_{k=-j}^{j} \frac{(j+l)!}{j!\, l!\, 2^{j+l+1}} \bar{O}_{lm} | \mathcal{L}_{l+j,m+k} | \bar{O}_{jk} \, .$$

Since the element g_{00} vanishes, we do not distinguish between the g_{00} and the other g_{jl} terms, despite the fact that the correction term is not valid for g_{00}. The energy $\bar{E}$ is then found to be

$$\bar{E}_{\text{lattice}} = \frac{1}{2} \sum_{l=0}^{\infty} \sum_{j=0}^{\infty} g_{jl} \, .$$

However, we need to introduce a truncation to proceed further, since an FMM implementation needs a finite order of poles p. Therefore, we split the infinite sums in the last equation as follows

$$\begin{aligned}
\bar{E}_{\text{lattice}} &= \frac{1}{2}\sum_{l=0}^{\infty}\sum_{j=0}^{\infty} g_{jl} \\
&= \underbrace{\frac{1}{2}\sum_{l=0}^{p}\sum_{j=0}^{p} g_{jl}}_{\text{Lattice Energy}} + \underbrace{\frac{1}{2}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty} g_{jl}}_{\text{Lattice Error}} + \underbrace{\frac{1}{2}\sum_{l=p+1}^{\infty}\sum_{j=0}^{p} g_{jl}}_{\text{Lattice Error}} + \underbrace{\frac{1}{2}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty} g_{jl}}_{\text{Lattice Error}} \\
&= \bar{E}_{\text{lattice}}(p) + \underbrace{\Delta\bar{E}_{A_1}(p) + \Delta\bar{E}_{A_2}(p) + \Delta\bar{E}_{B}(p)}_{\Delta\bar{E}_{\text{lattice}}(p)}
\end{aligned} \tag{3.7}$$

The lattice energy error reads

$$\begin{aligned}
\Delta\bar{E}_{\text{lattice}}(p) &= \frac{1}{2}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty} g_{jl} + \frac{1}{2}\sum_{l=p+1}^{\infty}\sum_{j=0}^{p} g_{jl} + \frac{1}{2}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty} g_{jl} \\
&= \frac{1}{2}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty} g_{jl} + \frac{1}{2}\sum_{l=p+1}^{\infty}\sum_{j=0}^{\infty} g_{jl}\,.
\end{aligned}$$

Again, we precompute the error terms $\Delta\bar{E}(p)$ up to order 50. To apply the scheme in an FMM computation, we have to calculate the total absolute charge for all particles N in the simulation box with

$$Q = \sum_{i=1}^{N} |q_i|\,.$$

The number of particles in box 1 and box 2 are identical, since each lattice image box is a replica of the original simulation box. The requested energy error couples to our precomputed error terms with

$$\Delta E_{\text{req}} \geq Q^2 \Delta\bar{E}_{\text{lattice}}(p)\,.$$

The order of poles p sufficient to satisfy the requested error bound is obtained similarly to the open boundary case. Since we reduced the interaction of the simulation box with the lattice to just one box-box interaction, we obtained a much simpler form for the final equation.

3.7.2. Stage II of the Error Scheme

In comparison to the second stage of the error scheme with open boundaries, we can not use the rotation-based approach $\mathcal{O}(p^3)$ for the convoluted lattice operator $\mathcal{L}$. Instead the $\mathcal{O}(p^4)$ operators are applied. Since this step does not depend on the number of particles, but the order of poles p, the $\mathcal{O}(p^4)$ overhead is still negligible.

The lattice energy is defined by

$$E_{\text{lattice}} = \frac{1}{2}\sum_{l=0}^{\infty}\sum_{m=-l}^{l}\sum_{j=0}^{\infty}\sum_{k=-j}^{j}(\pm 1)^{j+l}(-1)^{j}\omega_{lm}(\mathbf{a}_1)\,\mathcal{L}_{l+j,m+k}\,\omega_{jk}(\mathbf{a}_2)\,. \tag{3.8}$$

We define a charge multipole expansion as

$$O_{lm}(\mathbf{a}) = a^l \tilde{P}_{lm}(\cos\alpha)e^{-im\beta}\,.$$

Box-Box Representation of the Lattice Energy

Now we can replace the multipole moments in (3.8) with the chargeless version of box A with $O_{lm}(\mathbf{a}_1)$ and box B with $O_{jk}(\mathbf{a}_2)$. We obtain two energy formulas, depending on the relative position of the two boxes A and B. The energies read

$$E_{\text{A-B}} = \frac{1}{2}\sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\sum_{m=-l}^{l}\sum_{k=-j}^{j}(-1)^{j}O_{lm}(\mathbf{a}_1)\mathcal{L}_{l+j,m+k}O_{jk}(\mathbf{a}_2)$$
$$E_{\text{B-A}} = \frac{1}{2}\sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\sum_{m=-l}^{l}\sum_{k=-j}^{j}(-1)^{l}O_{lm}(\mathbf{a}_2)\mathcal{L}_{l+j,m+k}O_{jk}(\mathbf{a}_1)\,.$$

We label the inner sums of the first equation as

$$\hat{S}_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathbf{R}) = \sum_{m=-l}^{l}\sum_{k=-j}^{j}(-1)^{j}O_{lm}(\mathbf{a}_1)\mathcal{L}_{l+j,m+k}O_{jk}(\mathbf{a}_2)$$

and due to symmetry the inner sums of the second equation as

$$(-1)^{j+l}\hat{S}_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R}) = \sum_{m=-l}^{l}\sum_{k=-j}^{j}(-1)^{j}O_{lm}(\mathbf{a}_2)\mathcal{L}_{l+j,m+k}O_{jk}(\mathbf{a}_1)\,.$$

Since we do not want to store any information depending the relative position of $\mathbf{a}_1$ and $\mathbf{a}_2$, we introduce a general position-independent S_{lj} as

$$S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L}) = (\pm 1)^{j+l}\hat{S}_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})\,.$$

Following the scheme from the first stage, we now split the outer infinite sums as follows

$$\sum_{l=0}^{\infty}\sum_{j=0}^{\infty}\cdots = \sum_{l=0}^{p}\sum_{j=0}^{p}\cdots + \sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\cdots + \sum_{l=p+1}^{\infty}\sum_{j=0}^{p}\cdots + \sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty}\cdots\,.$$

Hence, some error terms from the interaction of box A with box B are identical to error terms from the interaction from box B to box A. The energy for the interaction of box A with box B becomes

$$\begin{aligned}\Delta E_{\text{A-B}} =& \frac{1}{2}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})+(-1)^{j+l}S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathcal{L})\Big]+\\ & \frac{1}{2}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty}S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})\,.\end{aligned}$$

The energy for the interaction of box B with box A yields

$$\begin{aligned}\Delta E_{\text{B-A}} =& \frac{1}{2}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[(-1)^{j+l}S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathcal{L})+S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})\Big]+\\ & \frac{1}{2}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty}(-1)^{j+l}S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathcal{L})\,.\end{aligned}$$

By adding the two energies $E_{\text{A-B}}$ and $E_{\text{B-A}}$, we obtain

$$\begin{aligned}\Delta E =& \frac{1}{2}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})+(-1)^{j+l}S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathcal{L})\Big]+\\ & \frac{1}{4}\sum_{l=p+1}^{\infty}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})+(-1)^{j+l}S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathcal{L})\Big]\,.\end{aligned}$$

To simplify the last double sum, we take all terms with $j \geq p+1$ into account and drop all terms $l \geq p+2$. The equation yields

$$\begin{aligned}\Delta E \approx& \frac{1}{2}\sum_{l=0}^{p}\sum_{j=p+1}^{\infty}\Big[S_{lj}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})+(-1)^{j+l}S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathcal{L})\Big]+\\ & \frac{1}{4}\sum_{j=p+1}^{\infty}\left[S_{p+1,j}(\mathbf{a}_1,\mathbf{a}_2,\mathcal{L})+(-1)^{j+p+1}S_{p+1,j}(\mathbf{a}_2,\mathbf{a}_1,\mathcal{L})\right]\,.\end{aligned} \tag{3.9}$$

Now, both sums contain contributions $S_{lj}(\mathbf{a}_2,\mathbf{a}_1,\mathbf{R})$ expressed as interactions from box B with box A. We therefore neglect these terms in the following procedure, but include interactions from box A and B and again the interaction from box B to A.

Following a similar scheme as proposed in the first stage of the error control, we perform the summation of (3.9) stepwise with respect to l. For all values $l < p+1$ only the first term of (3.9) needs to be considered. For the last term $l = p+1$ only the last term of (3.9) has a contribution.

Starting with the contribution from the first sum for $l = 0$ gives

$$\Delta E_{l=0} = \frac{1}{2} \sum_{j=p+1}^{\infty} S_{0j}(\mathbf{a}_1, \mathbf{a}_2, \mathcal{L}) \,.$$

It simplifies to the zeroth order energy with

$$\begin{aligned} \Delta E_{l=0}(p) &= \frac{1}{2} \left| \sum_{j=p+1}^{\infty} \sum_{k=-j}^{j} (\pm 1)^j \mathcal{L}_{jk} O_{jk}(\mathbf{a}_2) \right| \\ &= \frac{1}{2} \left| \sum_{j=p+1}^{\infty} \sum_{k=-j}^{j} (\pm 1)^j \mathcal{L}_{jk} a_2^j \tilde{P}_{jk}(\cos \alpha_2) e^{-ik\beta_2} \right| . \end{aligned}$$

Grid Box A and Radial Parts of Box B

We proceed and establish a grid in box A with 33^3 grid points (a_G, α_G, β_G) and reduce all coordinates in box B to their radial distance a_2 and positions for the angles α_2 and β_2 on the connecting lines of the box. The energy error $\Delta E(p)$ can now computed stepwise for each grid point starting with $l = 0$

$$F_{0p}(a_G, \alpha_G, \beta_G) = \max_{s \in \{-1,1\}} \left| \sum_{j=p+1}^{\infty} \sum_{k=-j}^{j} s^j a_G^j \tilde{P}_{jk}(\cos \alpha_G) e^{-ik\beta_G} \mathcal{L}_{jk} \right| .$$

For all higher elements F_{qp} with $q \leq p + 1$, we store the maximum contribution of all possible angles in the non-grid box as

$$F_{qp}(a_G, \alpha_G, \beta_G) = \max_{n \in \{1,\ldots,8\}} \max_{s \in \{-1,1\}} \left| \widetilde{F}_{qp}^n(a_G, \alpha_G, \beta_G) \right|$$

with

$$\widetilde{F}_{qp}^n(a_G, \alpha_G, \beta_G) = \sum_{m=-q}^{q} \sum_{j=p+1}^{\infty} \sum_{k=-j}^{j} s^j \tilde{P}_{qm}(\cos \alpha_{2_n}) e^{-im\beta_{2_n}} \mathcal{L}_{l+j,m+k} a_G^j P_{jk}(\cos \alpha_G) e^{-ik\beta_G} \,.$$

Again we only store the largest contribution for the error F_{qp} as

$$F_{qp}(a_G, \alpha_G, \beta_G) = \frac{1}{1 + \delta_{p+1-q,0}} \max_{n \in \{1,\ldots,8\}} \max_{s \in \{-1,1\}} \left| \widetilde{F}_{qp}^n(a_G, \alpha_G, \beta_G) \right| \,.$$

We follow the symmetry considerations in chapter 2, section 2.7 and define H_{qp} as

$$\begin{aligned} H_{0p}(x,y,z) &= \sum_{\text{perm}\{\text{x, y, z}\}} F_{0p}(x,y,z) \\ &\vdots \\ H_{qp}(x,y,z) &= \sum_{\text{perm}\{\text{x, y, z}\}} F_{qp}(x,y,z) \end{aligned}$$

with $x, y, z \in \{0, \pm\frac{1}{32}, \ldots, \pm\frac{29}{32}, \pm\frac{31}{32}\}$, $p \in \{0, \ldots, 50\}$ and $q \in \{0, \ldots, p+1\}$. Again, perm{x, y, z} represents all possible permutations of the coordinates x, y, z.

Radial Part of the Non-Grid Box B

The remaining radial part from box B with a_2 is handled via f_q functions

$$f_q = \sum_{i=1}^{N} |q_i| a_i^q$$

To link the error terms from the grid box $H_{qp}(x,y,z)$ with the radial part of the non-grid box, we set up a corresponding grid structure $Q_{qp}(G)$ storing all contributions from f_q of box 2 for any non-empty grid box (box 1). The elements f_q do not depend on p, therefore an element $Q_q(G)$ is valid for any p. We define

$$Q_q = q_G \cdot f_q$$

with $q_G = \sum_{i=1}^{N_G} |q_i|$ being the charge of grid box G.

Linking Box A with Box B

Since we derived the error terms for the grid and non-grid box separately, we now have to combine both contributions to obtain the optimal order of poles p_{lattice}. Therefore, we calculate elements T_q by

$$\begin{aligned} T_0(G) &= Q_0(G) \cdot H_{0p}(G) \\ T_1(G) &= Q_1(G) \cdot H_{1p}(G) \\ &\vdots \\ T_q(G) &= Q_q(G) \cdot H_{qp}(G) \\ T_{q+1}(G) &= Q_{q+1}(G) \cdot H_{q+1,p}(G)\,. \end{aligned}$$

Finally we add up all error contributions from the grid points G and the contributions stored for each different pole

$$\Delta E_{\text{req}} \geq \sum_{G} \sum_{i=0}^{q+1} T_i(G)\,. \tag{3.10}$$

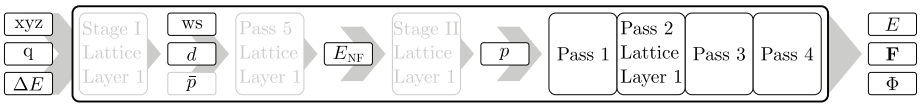
The order of poles p_{lattice} necessary for the computation of the lattice energy is performed by increasing q until the right hand side of (3.10) is less or equal to ΔE_{req}. Since we only use one order of multipoles in our code, we find the optimal order of poles for the simulation by

$$p = \max(p_{\text{lattice}}, p_{\text{box}})$$

with p_{lattice} being the order of poles necessary to compute the periodic lattice and p_{box} being the order of poles necessary to compute the contributions from the central simulation box.

Final FMM Parameter Set

After the error control stage I and II is conducted and we obtained the parameter set (ws, d, p), we can start the actual FMM calculation, computing the total energy, potentials and forces of the given particle system. This step does not differ from other FMM implementations and is described briefly in section 2.5 for open boundaries and again in section 3.6.2 for the additional steps with periodic boundaries.



In the next chapter we verify the scaling and precision of our error-controlled FMM implementation and compare against established competitor algorithms.

4. Performance Details of the FMM Implementation

Fast multipole methods have been available since 1987. However, the FMM rarely earned huge appreciation in the scientific community. Many scientific articles claimed that the advantage of the linear complexity will only be visible for very large systems with millions or even billions of particles [62]. These authors stated that large prefactors and the schemes to obtain the FMM parameter set will slow down calculations for a moderate number of particles. Since these statements strongly depend on the implementation and not on the FMM theory itself, in this chapter we will shed light on this claim and compare our error-controlled implementation against freely available codes from other groups dealing with long-range interactions. Thereby, we will show that the expected theoretical linear scaling along with high precision calculations are feasible – for open and mixed periodic boundary conditions as well.

4.1. Scaling

In the following section, we present the scaling of the derived algorithm with respect to the number of particles N. All tests were performed on JUMP, an IBM Power6 system with 4.7GHz, unless noted otherwise. We use five homogeneously distributed simulation sets. The smallest set contains only 4096 particles. Since the computation time for this configuration is around 0.1 seconds, an even smaller test case will not give any additional information concerning the scaling. The largest configuration contains about 16 million particles. All five sets listed in table 4.1, are used in a computation with mixed or full boundary conditions. The results are illustrated in figure 4.1.

Crossover Point with Direct Summation

The crossover point between an FMM and a direct summation can only be specified for open boundaries, since a direct summation is not possible for the periodic case due to the infinite

Periodicity	Number of Particles	Precision ΔE
Open, 1D, 2D, 3D	8^4 – 8^5 – 8^6 – 8^7 – 8^8	$1.00 \cdot 10^{-05}$

Table 4.1.: Number of particles N used in the scaling test.

number of particles in such systems. For open systems, the crossover point is determined at around 500 particles for low precision, i.e. $\Delta E = 10^{-3}$, and 4000 particles for high precision calculations, i.e. $\Delta E = 10^{-12}$. The computation time of the error control scheme is included therein.

Largest Test Set

Especially simulations in the field of astrophysics demand huge particle numbers. Therefore, we performed a computation with approximately 20 billion particles to check whether the implementation is capable of dealing with such an amount of data. The test set contained twice the number of particles used in the Millennium simulation [63]. The computation of the energy and forces took 2 days and 9 hours for one time step on a 1.6 GHz Itanium CPU with 1 TB of main memory. The precision was set to $\Delta E = 10^{-2}$, which is sufficient for most astrophysical calculations. This test example shows that this FMM implementation is capable of computing even very large particle ensembles with limited resources.

4.2. Precision Verification

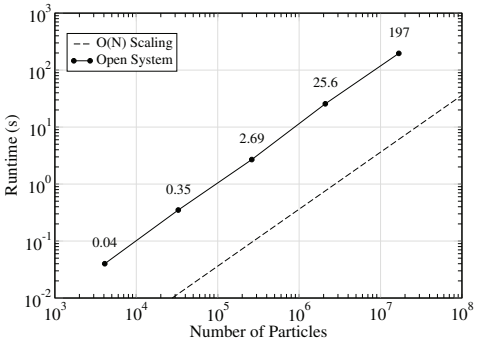
The precision of calculations with periodic boundaries can be verified directly against analytically known or rapidly converging solutions. That is, a Madelung particle system [64] serves as reference system. The Madelung constant characterizes the potential Φ at the origin due to the periodic lattice. The reference energy is determined by adding up all contributions $\frac{1}{2}\sum_{i=1}^{N} q_i \Phi(\mathbf{r}_i)$ inside the simulation box. The size of the simulation box for any periodicity can be increased by adding more and more particles from replica boxes. Thereby, it is possible to verify the precision of the algorithm even for millions of particles. All precision checks are performed for several simulation box sizes up to approximately 16 million particles. The results do not show any additional errors with increasing system size, except for minor fluctuations at machine precision for 8^8 particles. The data plotted in figure 4.2 is taken from test runs with the smallest possible Madelung particle sets, which can be constructed as follows.

Definition of 3D Periodic Systems

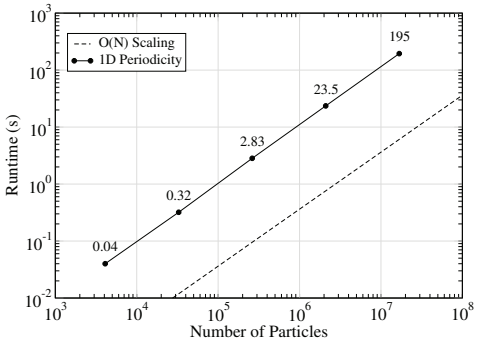
In three dimensions an ideal Madelung system consists of at least 8 particles as depicted in figure 4.3c. We define a simple periodic lattice by a three-dimensional Epstein zeta function [65], as in section 3.3. This function is given by

$$M^{\mathrm{3D}}(s) = \sum_{i,j,k\in\mathbb{Z}}{}' \frac{(-1)^{i+j+k}}{(i^2+j^2+k^2)^s} \,.$$

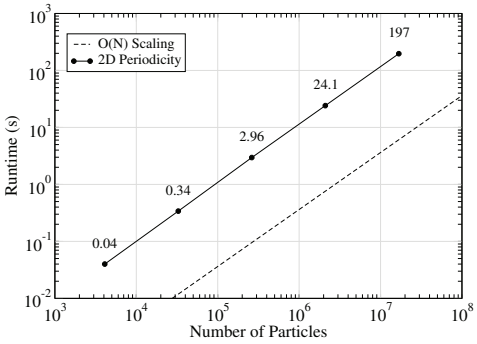
Unfortunately, no analytical solution for $M(s)$ is known. However, a rapidly converging double sum is available [66]. Since we are dealing with a Coulomb potential, the parameter s takes the



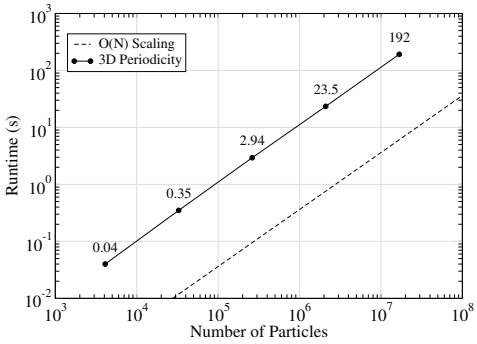
(a) Open boundaries, $\Delta E = 10^{-5}$



(b) 1D periodic boundaries, $\Delta E = 10^{-5}$



(c) 2D periodic boundaries, $\Delta E = 10^{-5}$



(d) 3D periodic boundaries, $\Delta E = 10^{-5}$

Figure 4.1.: Scaling of the FMM with open, 1D, 2D and 3D periodic boundary conditions. The solid black lines represent the actual periodicity. The remaining shaded lines show that the impact of the periodic boundary is limited to a factor of two, independent of the number of particles. All results show the optimal $\mathcal{O}(N)$ scaling compared to the reference scaling denoted by a dashed line. When increasing the number of particles for each plotted point by a factor of eight, the computation time increases roughly by a factor of eight. The difference in the runtime between open, 1D, 2D and 3D periodic systems originates from the increasing number of boxes at the boundary of the central box. For open systems, we do not have any interacting boxes across the boundaries and are allowed to apply a fractional tree depth. Therefore, the computation time for an open system will always be minimal compared to a periodic computation.

value $s = 1/2$. The double sum representation of $M(1/2)$ reads

$$M^{\mathrm{3D}}(1/2) = -12\pi \sum_{n=1}^{\infty}\sum_{p=1}^{\infty} \operatorname{sech}^2\left(\frac{\pi}{2}\left[(2n-1)^2 + (2p-1)^2\right]^{1/2}\right)$$
$$\approx -1.747\,564\,594\,633\,182\,190\,636\,212\,035\,544\,4\ldots .$$

Using the computer algebra system MAPLE [67], we compute 100 digits of this reference constant $M(1/2)$, which is sufficient for double and quadruple precision calculations.

Definition of 2D Periodic Systems

For 2D periodic systems, a Madelung counterpart [68] with four particles exists with

$$M^{\mathrm{2D}}(s) = \sum_{i,j\in\mathbb{Z}}{}' \frac{(-1)^{i+j}}{(i^2+j^2)^s}\,.$$

The constant $M(1/2)$ can be written in analytical form

$$M^{\mathrm{2D}}(1/2) = -4\beta(1/2)\cdot\eta(1/2)$$
$$\approx -1.615\,542\,626\,712\,824\,723\,867\,923\,332\,758\,6\ldots ,$$

where $\beta(s)$ denotes the Dirichlet beta function and $\eta(s)$ denotes the Dirichlet eta function [69].

Definition of 1D Periodic Systems

Madelung systems for 1D periodic systems consist of at least two particles. The generating function for such a periodic lattice is given by

$$M^{\mathrm{1D}}(s) = \sum_{i\in\mathbb{Z}}{}' \frac{(-1)^{i}}{i^{2s}}\,.$$

The reference constant $M(1/2)$ [70] reads

$$M^{\mathrm{1D}}(1/2) = -2\ln 2$$
$$\approx -1.386\,294\,361\,119\,890\,618\,834\,464\,242\,916\,4\ldots .$$

Verification of Precision and Order of Poles

In a first test, we increase the order of poles from 0 to 50 and check the achieved accuracy. The double precision (dp) test shows that 25 poles are sufficient to reach machine precision [71] $\varepsilon_{dp} = 2^{-52} \approx 2.22\times 10^{-16}$ for 1D and 2D periodic boundaries. The accuracy for 3D periodic systems is slightly lower, but still near machine precision. The quadruple precision

(qp) test shows similar results. To reach the machine precision limit $\varepsilon_{qp} = 2.47 \times 10^{-32}$, 40 to 50 multipoles are necessary. Again, for 3D periodic systems the ultimate accuracy is slightly lower. The results of these tests are illustrated in figures 4.2a and 4.2b. The target machine is an IBM Power6 system having only 105 significant bits for quadruple precision instead of 112 bits.

Verification of Precision and Energy Error Bound

In a second test, we do not adjust the order of poles manually. Instead we use our error-control scheme to obtain the optimal parameter set for runtime and precision. We increase the requested relative energy error, starting from $\Delta E = 1$ by an order of magnitude, until we pass machine precision $\Delta E = 1 \times 10^{-17}$. Again the calculation is performed in quadruple precision as well. Figures 4.2c and 4.2d show that the accuracy using the adjusted parameters does never exceed the user-requested error bound. The order of poles is minimal for almost every data point. A further reduction of multipoles for most data point would violate the requested error bound. That is, the error-control scheme obtains the minimal order of poles possible for the requested precision.

4.3. Benchmark

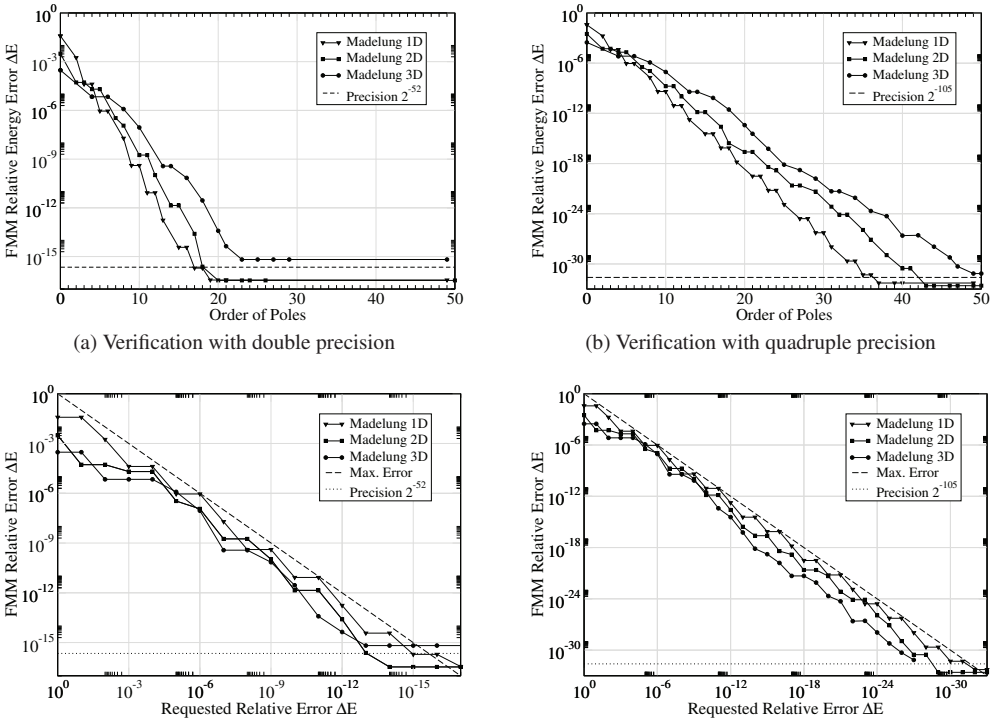
Finally, we compare our implementation against other available Coulomb solvers. The results of this section are taken from a performance comparison conducted at JSC [72]. For this benchmark only the algorithmic performance without any additional parallelization is reviewed. The consumable memory is limited to 6 GB. A full comparison with a larger variety of test cases and precision thresholds can be found in [73]. In this work we will limit ourselves to the case with periodic boundary conditions.

Test Case

The test case consists of an even number of randomly distributed particles inside a unit box with charge $q_i = \pm 1$. To ensure convergence of the periodic system, the total charge must be $Q = \sum_i^N q_i = 0$. The total number of particles is limited to 46656, since the competitor implementations exceed the provided resources for larger test sets. By contrast, the FMM implementation itself can handle several million particles with the provided resources.

Competitors

We compare our implementation against codes from the ESPRESSO software package [74]. Namely, P^{3}M for 3D periodic systems, MMM2D and ELC/P3M for 2D periodic systems and MMM1D for 1D periodic systems. The asymptotical complexity is composed in table 4.2 Each method includes an automatic tuning procedure for the forces. Since we compare timings for a given relative energy error, we conduct several runs of these codes with different force error



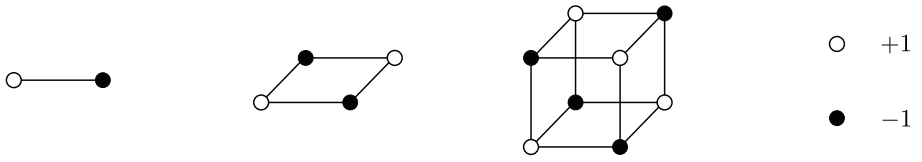
(a) Verification with double precision

(b) Verification with quadruple precision

(c) Energy error bound with double precision

(d) Energy error bound with quadruple precision

Figure 4.2.: Precision of FMM calculations with periodic boundary conditions. Approximately 25 poles are sufficient to obtain results in the range of double precision with 52 mantissa bits (a). For quadruple precision with 105 mantissa bits, 50 poles are sufficient to obtain the maximum precision (b). The requested energy error bounds, pictured as dashed lines in (c) and (d) are never exceeded.



(a) Madelung 1D

(b) Madelung 2D

(c) Madelung 3D

Figure 4.3.: All reference Madelung systems consist of alternating charges $q \in \{-1, +1\}$. The illustrated simulation box is replicated along the black lines connecting the charges $+1$ and -1, representing an infinite number of particles on a line (a), in a plane (b) or in the $\mathbb{R}^3$ volume (c), respectively.

Method	Periodicity	Complexity	Error Bound
FMM	1D, 2D, 3D	$\mathcal{O}(N)$	energy
MMM1D [75]	1D	$\mathcal{O}(N^2)$	force
MMM2D [76]	2D	$\mathcal{O}(N^{5/3})$	force
ELC/P^{3}M [77]	2D	$\mathcal{O}(N \log N)$	force
P^{3}M [78]	3D	$\mathcal{O}(N \log N)$	force

Table 4.2.: Asymptotic complexity and error bound measure of the compared Coulomb solvers.

			Timings t $[s]$		FMM Timings t $[s]$	
Method	Test Case	N	Tuning	Solving	Tuning	Solving
MMM1D	1D Madelung	512	0.6	0.06	0.005	0.002
MMM1D	1D Madelung	4096	43.6	3.54	0.018	0.011
MMM1D	Cube	4096	89.7	4.34	0.041	0.033
ELC/P^{3}M	2D Madelung	576	9.9	0.03	0.003	0.001
ELC/P^{3}M	2D Madelung	4096	328.0	0.46	0.013	0.013
P^{3}M	3D Madelung	512	73.3	0.04	0.003	0.003
P^{3}M	3D Madelung	4096	311.8	0.32	0.023	0.021
P^{3}M	Cube	4096	429.1	0.34	0.039	0.065

Table 4.3.: Tuning and solving time of established Coulomb solver compared to the overall FMM runtime. Even for small particle systems the tuning time of other codes are substantial. The FMM and the inherent error-control scheme provide the lowest computation times including error control.

bounds. Then, we compute the corresponding energy error bounds, and use the results to set up the energy error bound for the FMM. Due to the fact, that the tuning procedures of MMM1D, MMM2D, ELC and P^{3}M are not performed in every time step of the computation, we exclude the runtime of the parameter estimation scheme as well. However, for the FMM we do not need to exclude the runtime of the error control scheme, since we do have a small overhead of the error control scheme even for few particles. The runtime of the automatic tuning procedure and the runtime and tuning time of the FMM can be found in table 4.3.

Results

The runtime of the different algorithms is reviewed in [73] for several energy error bounds, starting from $\Delta E = 10^{-3}$ up to $\Delta E = 10^{-11}$. We present the results for a medium precision of $\Delta E = 10^{-5}$ for two- and three-dimensional periodicities and high precision $\Delta E = 10^{-11}$ for one-dimensional periodicity only, since the left out runs qualitatively show the same behavior.

For three-dimensional periodic systems up to 1500 particles, both algorithms roughly consume the same runtime. For larger particle systems, the FMM outperforms P^{3}M. In addition, the P^{3}M method does not reach a higher error bound than $\Delta E = 10^{-5}$.

For two-dimensional periodic systems two competitors, namely MMM2D and ELC/P^{3}M, are available. MMM2D is comparable in speed to the FMM for very small particle systems up to 256 particles. ELC/P^{3}M does perform better and is comparable to the FMM up to 1000 particles. For larger test cases with more particles the FMM exploits its linear complexity and performs significantly faster.

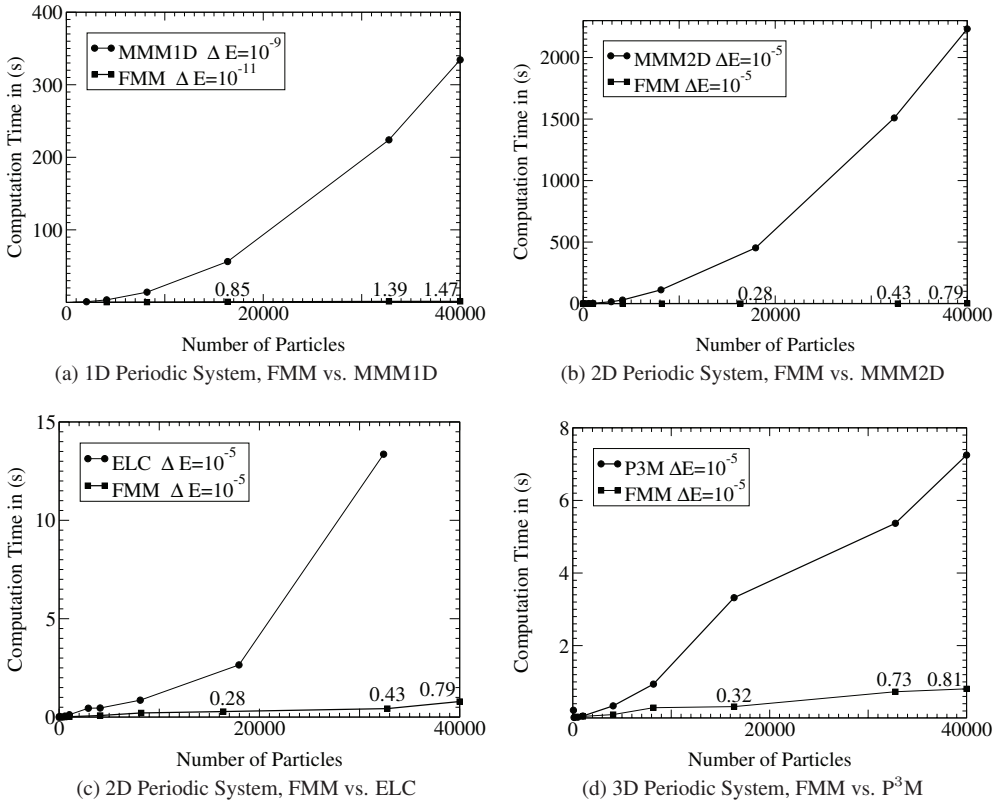
The last test case for one-dimensional periodicity shows similar results. MMM1D consumes roughly the same amount of time for up to 400 particles. Larger particle systems suffer from the poor $\mathcal{O}(N^2)$ complexity of the MMM1D algorithm.

The results for large particle systems are shown in figure 4.4. Finally, we can state that the optimal theoretical complexity of the FMM can be utilized for N-body simulations and allows to increase the number of particles significantly in a cubic simulation box for any kind of periodicity. For any given precision, the FMM performs fastest for all system sizes starting with only a few hundred particles.

4.4. Additional Features

The FMM is optimized to handle particle systems from several fields of research. Since the requirements for a Coulomb solver, such as the FMM, differ for each scientific problem, not all features can be discussed in the scope of this work. Therefore, we list the most important additional features that were not mentioned in this thesis briefly. The majority of enhancements were implemented to simulate highly clustered particle systems. The list starts with the most general enhancements, which will affect homogenously distributed particle systems as well and ends with features necessary to perform calculations in the field of laser-plasma physics.

- Available for single, double and quadruple precision
- Alignment of data for linear memory access
- Memory management preventing memory fragmentation for clustered systems
- Memory scaling with N, instead of tree depth L_{max} or order of poles p
- Memory footprint below 50 Bytes per particle for $p = 2$
- Cache optimization for near and far field operators
- CPU optimization: far field operators utilize available FMAs
- Detachable FMM operators: M2M, M2L, L2L
- Supplemental P2L treecode operator for boxes with few particles
- Box management overhead below 1% of the computation time
- Interaction list compilation as sorted list without search
- Detachable error control for fast low precision calculations
- Interchangeable near field potentials: Coulomb potential or linear cusp potential



(a) 1D Periodic System, FMM vs. MMM1D

(b) 2D Periodic System, FMM vs. MMM2D

(c) 2D Periodic System, FMM vs. ELC

(d) 3D Periodic System, FMM vs. P^3M

Figure 4.4.: Comparison of the FMM against MMM1D, MMM2D, ELC and P^3M. For homogeneous particle distributions the FMM outperforms other fast summation schemes. Especially for 1D and 2D periodic simulations, the optimal $\mathcal{O}(N)$ complexity of the FMM provides a substantial improvement over other algorithms.

5. Conclusion and Future Directions

5.1. Conclusion

With early methods and approaches, the solution of the N-body problem posed severe limitations to the number of particles used in a simulation. Due to the $\mathcal{O}(N^2)$ complexity, large particle ensembles could not be studied in a reasonable time period. With the advent of fast summation methods in the 1960s to 1980s, those limitations were pushed to larger particle numbers. However, most of the developed schemes lack an a priori error bound. In 1987, Greengard and Rokhlin proposed the Fast Multipole Method with optimal $\mathcal{O}(N)$ complexity and rigorous error bounds. Unfortunately, the method comprises three additional parameters, which have to be tuned in advance to obtain the desired scaling and precision. No analytical and still efficient error bounds for arbitrary particle systems are known, so far. Therefore, the tuning of the parameter set has to be performed on-the-fly for every simulation step. Different approaches were taken to circumvent this issue, one of them is described in this thesis.

In chapter 2, we introduced a uniform derivation of the FMM algorithm using a much simpler notation than originally proposed. A two-stage error control scheme developed by Dachsel [50] is presented as well. However, the first stage implies a correction factor, which has not been proven to maintain the error bound. The second stage implies an assumption on error contribution of the interaction set. Both inconsistencies could be resolved. It is shown analytically that both assumptions are valid and do not violate the error bound. In addition, for a homogeneous systems, the minimal order of poles necessary for a certain precision was derived analytically.

In chapter 3, the FMM was successfully extended to periodic boundary conditions in one, two and three dimensions. The universal lattice constants $\mathcal{L}_{lm}$ were derived for one-, two- and three dimensional cubic lattices in high precision. The requirements to preserve the optimal $\mathcal{O}(N)$ scaling and to connect the error control scheme for open boundaries were met. We derived the necessary modifications along with the new error control scheme for periodic boundaries. The optimal $\mathcal{O}(N)$ complexity was maintained even for mixed boundary conditions.

In the last chapter, we performed the necessary benchmark tests to show that the implemented structures still achieve linear scaling. The error estimation scheme was tested for several precisions with one-, two- and three-dimensional periodicity and revealed tight error bounds never exceeding the requested threshold. The comparison against other Coulomb solver codes showed superior scaling properties even for a small number of particles.

Finally, we can conclude that the presented FMM implementation provides an ideal Coulomb solver for open and periodic boundary conditions. The scientific community may benefit from this implementation, since algorithmic details, like tuning the FMM parameters are hidden from

the user and will be adjusted automatically without affecting the computation time. Even high precision calculations up to machine precision can be performed without degrading the runtime. Furthermore, the developed FMM library is independent of third-party libraries, say FFTW or BLAS, and has a small memory footprint.

The developed code provides a firm framework for simulations in the field of molecular dynamics and astrophysics. It is well tested with real world examples and is used in a production environment at Max Planck Institute for the Physics of Complex Systems in Dresden [79, 80, 81, 82].

5.2. Future Directions

The FMM and especially the presented implementation offers a wide range of possibilities for future enhancements. The following list depicts possible future directions for this implementation:

- Full parallelization of the code [83, 84]
- Extension to general potentials of the form r^{-n} [85]
- Extension to one-component plasma calculations [86]
- Extension to periodicities with non-cubic simulation boxes
- Error control for the forces and the potential
- Extension to systems with dipoles instead of charges [87]

The ideal scaling and the superior sequential performance of the FMM are indispensable requirements for all future directions. The parallelization of the FMM is the most vital issue and will be addressed within the ScaFaCoS project [88].

Acknowledgements

First of all I would like to express my sincere gratitude to my advisor Prof. Dr. Bruno Lang for his supervision and guidance. His help and encouragement enabled me to focus on the main target points of this thesis.

A special thanks goes to Dr. Holger Dachsel, who is most responsible for helping me understand the labyrinthine insights of the Fast Multipole Method. His thorough and foresighted work inspired me to extend the error control scheme further. The endless but fruitful discussions about additional features and optimizations of the code have proven worthwhile.

Besides Holger Dachsel I like to thank the many colleagues at Jülich Supercomputing Centre for their assistance. Especially Prof. Dr. Dr. Thomas Lippert, who gave me the possibility to work in his institute on this thesis.

Let me also say 'thank you' to the former and current PhD students at JSC. It was a great pleasure for me to be part of the many philosophical, economical and scientific discussions. These happenings I could not take for granted, considering the fact that my workday was somehow synchronized with Atlantic Daylight Time (ADT), not Central European Time (CET).

A. Appendix

l	m	$\bar{O}_{lm}$
0	0	$1.000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,0 \times 10^{+00}$
2	0	$0.000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,000\,0 \times 10^{+00}$
4	0	$-6.076\,388\,888\,888\,888\,888\,888\,888\,888\,888\,888\,888\,888\,888\,888\,888\,9 \times 10^{-04}$
4	4	$-4.340\,277\,777\,777\,777\,777\,777\,777\,777\,777\,777\,777\,777\,777\,777\,777\,8 \times 10^{-05}$
6	0	$2.066\,798\,941\,798\,941\,798\,941\,798\,941\,798\,941\,798\,941\,798\,941\,798\,941\,8 \times 10^{-06}$
6	4	$-1.033\,399\,470\,899\,470\,899\,470\,899\,470\,899\,470\,899\,470\,899\,470\,899\,470\,9 \times 10^{-06}$
8	0	$2.664\,233\,010\,912\,698\,412\,698\,412\,698\,412\,698\,412\,698\,412\,698\,412\,698\,4 \times 10^{-08}$
8	4	$3.767\,602\,237\,654\,320\,987\,654\,320\,987\,654\,320\,987\,654\,320\,987\,654\,320\,987\,7 \times 10^{-09}$
8	8	$1.345\,572\,227\,733\,686\,067\,019\,400\,352\,733\,686\,067\,019\,400\,352\,733\,686\,067\,7 \times 10^{-10}$
10	0	$-1.060\,147\,815\,790\,176\,901\,288\,012\,399\,123\,510\,234\,621\,345\,732\,5 \times 10^{-10}$
10	4	$4.892\,989\,919\,031\,585\,698\,252\,364\,919\,031\,585\,698\,252\,364\,919\,0 \times 10^{-11}$
10	8	$4.077\,491\,599\,192\,988\,081\,876\,970\,765\,859\,654\,748\,543\,637\,432\,5 \times 10^{-12}$
12	0	$-1.399\,537\,380\,973\,005\,352\,965\,670\,425\,987\,886\,305\,346\,622\,806\,9 \times 10^{-13}$
12	4	$-1.234\,134\,326\,318\,241\,713\,827\,031\,287\,348\,747\,666\,207\,983\,668\,3 \times 10^{-13}$
12	8	$1.424\,741\,655\,968\,016\,955\,120\,129\,723\,304\,326\,478\,929\,653\,532\,8 \times 10^{-14}$
12	12	$-8.751\,484\,373\,267\,917\,414\,742\,811\,568\,208\,393\,605\,219\,002\,044\,4 \times 10^{-17}$
14	0	$9.918\,348\,956\,370\,306\,403\,375\,186\,443\,969\,512\,752\,581\,535\,650\,3 \times 10^{-16}$
14	4	$-4.171\,540\,884\,591\,040\,634\,360\,740\,180\,846\,000\,951\,821\,057\,641\,2 \times 10^{-16}$
14	8	$-7.584\,619\,790\,165\,528\,426\,110\,436\,692\,447\,274\,457\,856\,468\,438\,5 \times 10^{-17}$
14	12	$-2.917\,161\,457\,755\,972\,471\,580\,937\,189\,402\,797\,868\,406\,334\,014\,8 \times 10^{-18}$
16	0	$-6.898\,541\,667\,295\,929\,079\,355\,195\,750\,828\,077\,832\,953\,857\,437\,7 \times 10^{-19}$
16	4	$1.141\,750\,539\,915\,212\,877\,587\,208\,106\,452\,841\,383\,850\,824\,664\,2 \times 10^{-18}$
16	8	$-1.686\,930\,836\,861\,028\,811\,307\,910\,215\,991\,967\,210\,973\,331\,940\,2 \times 10^{-19}$
16	12	$-1.802\,372\,019\,957\,478\,906\,562\,976\,919\,514\,596\,313\,711\,021\,322\,5 \times 10^{-20}$
16	16	$1.861\,954\,566\,071\,775\,729\,920\,430\,701\,977\,888\,753\,833\,699\,713\,3 \times 10^{-23}$
18	0	$-2.349\,516\,002\,320\,344\,973\,685\,058\,771\,508\,339\,423\,007\,997\,823\,7 \times 10^{-21}$
18	4	$6.727\,302\,512\,403\,688\,953\,015\,771\,679\,126\,126\,384\,778\,600\,267\,4 \times 10^{-22}$
18	8	$5.039\,130\,372\,498\,612\,291\,221\,247\,842\,666\,173\,914\,636\,045\,860\,6 \times 10^{-22}$
18	12	$-2.538\,604\,721\,661\,769\,416\,232\,366\,671\,368\,349\,579\,161\,735\,949\,9 \times 10^{-24}$
18	16	$6.533\,173\,916\,041\,318\,350\,598\,002\,463\,080\,311\,416\,960\,349\,871\,2 \times 10^{-25}$
20	0	$3.934\,798\,872\,544\,174\,981\,882\,536\,980\,622\,681\,026\,062\,887\,993\,1 \times 10^{-24}$
20	4	$-3.382\,538\,200\,640\,741\,116\,500\,133\,088\,363\,849\,242\,836\,078\,872\,4 \times 10^{-24}$
20	8	$4.224\,158\,289\,099\,523\,672\,993\,184\,912\,360\,551\,190\,718\,498\,943\,7 \times 10^{-25}$
20	12	$1.189\,225\,242\,839\,065\,363\,215\,210\,153\,099\,716\,893\,809\,165\,959\,3 \times 10^{-25}$
20	16	$4.974\,783\,912\,326\,511\,183\,161\,323\,912\,887\,351\,581\,197\,993\,687\,4 \times 10^{-27}$
20	20	$-1.657\,156\,533\,086\,779\,208\,248\,275\,787\,104\,380\,939\,772\,816\,018\,5 \times 10^{-30}$

Table A.1.: Averaged chargeless multipole moments $\bar{O}_{lm}$ for a homogenously distributed particle system inside a unit box with total charge $Q = \sum_{i=1}^{\infty} q_i = 1$.

l	m	$\mathcal{L}_{lm}$
2	0	$8.082\,276\,126\,383\,771\,415\,989\,526\,460\,457\,999\,630\,599\,451\,693\,6 \times 10^{-01}$
4	0	$1.772\,532\,246\,881\,756\,463\,905\,543\,349\,937\,640\,066\,739\,884\,136\,1 \times 10^{+00}$
6	0	$1.202\,295\,942\,996\,887\,064\,930\,847\,178\,370\,733\,382\,380\,352\,721\,4 \times 10^{+01}$
8	0	$1.619\,567\,974\,952\,697\,706\,556\,473\,109\,017\,085\,575\,592\,558\,564\,8 \times 10^{+02}$
10	0	$3.586\,623\,213\,257\,425\,981\,237\,685\,664\,108\,210\,914\,558\,068\,158\,8 \times 10^{+03}$
12	0	$1.175\,597\,796\,629\,048\,537\,535\,289\,981\,272\,694\,379\,418\,371\,282\,4 \times 10^{+05}$
14	0	$5.333\,260\,344\,135\,690\,382\,440\,620\,728\,715\,132\,212\,909\,659\,250\,3 \times 10^{+06}$
16	0	$3.195\,829\,630\,218\,132\,633\,113\,762\,230\,287\,582\,205\,028\,306\,347\,5 \times 10^{+08}$
18	0	$2.443\,418\,184\,280\,147\,129\,890\,664\,795\,105\,989\,437\,485\,149\,751\,8 \times 10^{+10}$
20	0	$2.320\,662\,442\,643\,474\,513\,636\,486\,394\,994\,172\,927\,521\,838\,413\,5 \times 10^{+12}$
22	0	$2.680\,065\,671\,012\,043\,058\,980\,038\,886\,253\,743\,102\,923\,181\,183\,6 \times 10^{+14}$
24	0	$3.698\,307\,224\,343\,398\,304\,098\,927\,152\,679\,744\,744\,623\,336\,748\,3 \times 10^{+16}$
26	0	$6.009\,616\,888\,311\,914\,822\,177\,647\,475\,607\,803\,703\,572\,102\,050\,5 \times 10^{+18}$
28	0	$1.135\,806\,479\,353\,612\,288\,448\,329\,536\,880\,984\,569\,532\,259\,100\,9 \times 10^{+21}$
30	0	$2.470\,368\,353\,169\,561\,293\,910\,902\,500\,768\,263\,913\,222\,759\,431\,9 \times 10^{+23}$
32	0	$6.126\,501\,680\,296\,544\,447\,284\,919\,102\,519\,967\,395\,502\,570\,761\,6 \times 10^{+25}$
34	0	$1.718\,482\,245\,939\,763\,174\,354\,527\,632\,056\,909\,442\,250\,425\,840\,8 \times 10^{+28}$
36	0	$5.413\,217\,009\,265\,466\,591\,467\,282\,132\,417\,883\,654\,079\,081\,125\,7 \times 10^{+30}$
38	0	$1.902\,745\,456\,096\,518\,157\,221\,254\,903\,838\,890\,293\,287\,921\,183\,6 \times 10^{+33}$
40	0	$7.420\,706\,719\,506\,456\,303\,035\,205\,746\,047\,194\,586\,644\,117\,231\,3 \times 10^{+35}$
42	0	$3.194\,614\,135\,741\,390\,559\,440\,881\,591\,279\,786\,027\,707\,180\,006\,8 \times 10^{+38}$
44	0	$1.511\,052\,463\,710\,709\,654\,471\,879\,559\,629\,321\,548\,825\,933\,854\,4 \times 10^{+41}$
46	0	$7.819\,696\,447\,964\,625\,537\,355\,713\,582\,359\,284\,207\,854\,004\,258\,6 \times 10^{+43}$
48	0	$4.410\,308\,783\,683\,000\,655\,281\,503\,546\,993\,278\,137\,255\,706\,817\,7 \times 10^{+46}$
50	0	$2.701\,314\,126\,475\,377\,593\,713\,915\,529\,321\,422\,882\,868\,073\,899\,6 \times 10^{+49}$
52	0	$1.790\,971\,264\,812\,866\,837\,947\,624\,571\,469\,202\,318\,946\,523\,599\,0 \times 10^{+52}$
54	0	$1.281\,439\,939\,642\,788\,200\,417\,250\,328\,168\,534\,251\,952\,863\,196\,3 \times 10^{+55}$
56	0	$9.867\,087\,534\,117\,336\,516\,110\,792\,879\,668\,841\,474\,536\,195\,295\,0 \times 10^{+57}$
58	0	$8.155\,147\,846\,532\,108\,611\,886\,504\,180\,353\,586\,635\,498\,965\,132\,1 \times 10^{+60}$
60	0	$7.217\,305\,844\,017\,340\,588\,142\,510\,100\,595\,776\,181\,220\,242\,913\,7 \times 10^{+63}$

Table A.2.: Non-zero high precision coefficients $\mathcal{L}_{lm}$ for cubic 1D periodic systems.

l	m	$\mathcal{L}_{lm}$
2	0	$1.797\,422\,737\,979\,606\,871\,425\,168\,454\,745\,963\,191\,097\,381\,805\,2 \times 10^{+00}$
4	0	$2.349\,346\,404\,583\,329\,186\,258\,405\,792\,818\,084\,553\,190\,924\,469\,0 \times 10^{+00}$
4	4	$1.604\,987\,200\,845\,380\,002\,407\,942\,643\,682\,803\,942\,848\,481\,601\,8 \times 10^{+01}$
6	0	$8.882\,263\,548\,147\,084\,246\,240\,135\,691\,775\,708\,527\,592\,857\,769\,2 \times 10^{+00}$
6	4	$3.220\,589\,033\,162\,925\,356\,576\,340\,186\,273\,781\,478\,191\,205\,872\,6 \times 10^{+00}$
8	0	$1.380\,144\,487\,047\,449\,672\,641\,392\,183\,572\,404\,509\,962\,469\,710\,9 \times 10^{+02}$
8	4	$1.227\,678\,963\,816\,241\,808\,486\,636\,372\,422\,657\,735\,077\,862\,293\,0 \times 10^{+02}$
8	8	$1.078\,667\,598\,637\,292\,590\,273\,066\,840\,357\,080\,405\,868\,758\,726\,7 \times 10^{+04}$
10	0	$2.165\,806\,347\,210\,116\,496\,745\,264\,250\,097\,006\,229\,232\,175\,477\,8 \times 10^{+03}$
10	4	$1.546\,663\,515\,968\,671\,757\,393\,495\,334\,711\,681\,848\,420\,816\,663\,0 \times 10^{+01}$
10	8	$-9.785\,756\,849\,360\,865\,728\,339\,599\,084\,119\,378\,786\,137\,775\,666\,7 \times 10^{+03}$
12	0	$1.598\,019\,973\,372\,856\,838\,618\,981\,311\,669\,383\,509\,347\,273\,036\,1 \times 10^{+05}$
12	4	$1.103\,683\,169\,310\,599\,243\,246\,141\,080\,798\,803\,618\,309\,098\,706\,3 \times 10^{+05}$
12	8	$4.303\,842\,787\,865\,130\,279\,920\,729\,264\,630\,707\,980\,343\,456\,665\,7 \times 10^{+05}$
12	12	$8.736\,315\,496\,702\,659\,216\,015\,096\,919\,670\,210\,809\,673\,623\,865\,5 \times 10^{+07}$
14	0	$4.392\,284\,573\,648\,583\,640\,732\,888\,271\,441\,448\,519\,223\,714\,611\,4 \times 10^{+06}$
14	4	$-1.888\,692\,716\,650\,118\,963\,136\,673\,285\,503\,138\,243\,957\,898\,221\,7 \times 10^{+06}$
14	8	$-1.450\,922\,800\,380\,728\,348\,289\,821\,337\,066\,626\,085\,326\,879\,494\,9 \times 10^{+07}$
14	12	$-2.725\,386\,102\,343\,095\,947\,152\,371\,803\,830\,225\,020\,098\,715\,040\,5 \times 10^{+08}$
16	0	$4.103\,912\,081\,964\,717\,008\,539\,452\,068\,006\,723\,436\,733\,880\,556\,9 \times 10^{+08}$
16	4	$1.196\,930\,011\,890\,428\,992\,944\,079\,421\,078\,321\,668\,930\,933\,156\,2 \times 10^{+08}$
16	8	$7.783\,019\,100\,968\,639\,615\,889\,981\,196\,228\,172\,732\,465\,170\,756\,4 \times 10^{+08}$
16	12	$1.090\,776\,974\,897\,381\,759\,526\,168\,866\,801\,397\,999\,819\,363\,437\,0 \times 10^{+10}$
16	16	$3.079\,716\,907\,636\,877\,904\,110\,948\,574\,642\,234\,133\,795\,136\,946\,1 \times 10^{+12}$
18	0	$2.079\,412\,301\,782\,584\,720\,309\,192\,850\,084\,927\,791\,580\,008\,931\,4 \times 10^{+10}$
18	4	$-8.143\,684\,568\,934\,241\,063\,913\,438\,944\,481\,334\,294\,725\,470\,756\,6 \times 10^{+09}$
18	8	$-2.098\,504\,440\,998\,830\,120\,353\,196\,600\,977\,688\,698\,563\,700\,166\,4 \times 10^{+10}$
18	12	$-4.436\,726\,028\,931\,628\,068\,530\,826\,894\,912\,220\,990\,370\,536\,235\,8 \times 10^{+11}$
18	16	$-1.831\,589\,839\,076\,762\,591\,341\,771\,241\,083\,797\,228\,086\,312\,891\,1 \times 10^{+13}$
20	0	$2.557\,923\,832\,247\,806\,212\,364\,402\,821\,092\,621\,286\,746\,854\,753\,1 \times 10^{+12}$
20	4	$3.045\,350\,350\,032\,337\,515\,301\,314\,442\,208\,313\,797\,356\,752\,799\,1 \times 10^{+11}$
20	8	$2.506\,641\,138\,228\,522\,260\,209\,689\,249\,113\,707\,404\,717\,147\,464\,8 \times 10^{+12}$
20	12	$2.092\,539\,153\,353\,154\,546\,511\,320\,346\,340\,116\,199\,344\,909\,319\,0 \times 10^{+13}$
20	16	$7.457\,337\,726\,846\,175\,387\,435\,174\,097\,071\,966\,176\,029\,537\,742\,3 \times 10^{+14}$
20	20	$3.113\,812\,187\,187\,460\,002\,648\,484\,847\,731\,479\,054\,971\,092\,500\,7 \times 10^{+17}$

Table A.3.: Non-zero high precision coefficients $\mathcal{L}_{lm}$ for cubic 2D periodic systems.

l	m	$\mathcal{L}_{lm}$
4	0	$2.811\,930\,487\,188\,866\,820\,620\,048\,187\,991\,885\,560\,411\,823\,167\,8 \times 10^{+00}$
4	4	$1.405\,965\,243\,594\,433\,410\,310\,024\,093\,995\,942\,772\,477\,187\,316\,0 \times 10^{+01}$
6	0	$5.479\,590\,873\,932\,164\,406\,932\,770\,290\,083\,406\,409\,437\,389\,876\,9 \times 10^{-01}$
6	4	$-3.835\,713\,611\,752\,515\,084\,852\,939\,203\,058\,384\,603\,676\,375\,428\,1 \times 10^{+00}$
8	0	$1.215\,615\,730\,209\,791\,894\,211\,594\,848\,276\,170\,559\,610\,679\,570\,7 \times 10^{+02}$
8	4	$1.215\,615\,730\,209\,791\,894\,211\,594\,848\,276\,170\,560\,410\,027\,970\,6 \times 10^{+02}$
8	8	$7.901\,502\,246\,363\,647\,312\,375\,366\,513\,795\,108\,615\,853\,074\,765\,8 \times 10^{+03}$
10	0	$3.117\,991\,673\,610\,912\,310\,782\,258\,727\,428\,012\,496\,691\,124\,166\,1 \times 10^{+02}$
10	4	$-6.859\,581\,681\,944\,007\,083\,720\,969\,200\,341\,627\,518\,609\,716\,215\,7 \times 10^{+02}$
10	8	$-1.166\,128\,885\,930\,481\,204\,232\,564\,764\,058\,076\,677\,748\,007\,657\,7 \times 10^{+04}$
12	0	$2.424\,561\,274\,735\,909\,221\,719\,964\,051\,649\,346\,904\,269\,588\,800\,5 \times 10^{+05}$
12	4	$2.037\,585\,826\,414\,026\,651\,016\,255\,763\,388\,558\,102\,533\,054\,445\,9 \times 10^{+05}$
12	8	$7.068\,266\,654\,598\,500\,070\,164\,463\,510\,778\,812\,526\,628\,613\,039\,2 \times 10^{+05}$
12	12	$2.370\,243\,598\,452\,707\,828\,763\,961\,791\,327\,126\,302\,592\,553\,616\,2 \times 10^{+08}$
14	0	$2.095\,408\,711\,988\,554\,264\,871\,397\,928\,640\,220\,756\,684\,684\,227\,0 \times 10^{+06}$
14	4	$-2.694\,096\,915\,413\,855\,483\,406\,083\,051\,108\,855\,258\,993\,126\,906\,3 \times 10^{+06}$
14	8	$-1.706\,261\,379\,762\,108\,472\,823\,852\,599\,035\,608\,330\,664\,088\,231\,4 \times 10^{+07}$
14	12	$-6.540\,668\,622\,421\,415\,812\,491\,434\,962\,969\,831\,934\,105\,847\,327\,5 \times 10^{+08}$
16	0	$5.427\,985\,829\,965\,016\,962\,438\,288\,507\,697\,746\,506\,181\,830\,989\,5 \times 10^{+08}$
16	4	$2.284\,104\,152\,910\,541\,287\,238\,348\,694\,933\,442\,027\,166\,901\,643\,4 \times 10^{+08}$
16	8	$1.297\,330\,185\,489\,575\,858\,291\,814\,405\,833\,207\,725\,597\,430\,834\,1 \times 10^{+09}$
16	12	$2.588\,248\,490\,005\,557\,563\,835\,534\,374\,165\,006\,875\,180\,041\,147\,9 \times 10^{+10}$
16	16	$6.997\,365\,354\,798\,420\,565\,674\,532\,024\,803\,329\,188\,583\,401\,105\,4 \times 10^{+12}$
18	0	$1.468\,604\,995\,125\,845\,081\,063\,298\,450\,987\,009\,285\,618\,619\,951\,4 \times 10^{+10}$
18	4	$-1.537\,634\,648\,799\,471\,257\,606\,140\,581\,789\,142\,653\,759\,367\,670\,3 \times 10^{+10}$
18	8	$-2.422\,692\,155\,856\,961\,414\,158\,055\,213\,390\,241\,021\,840\,216\,524\,7 \times 10^{+10}$
18	12	$-1.069\,241\,660\,473\,865\,905\,615\,256\,775\,112\,719\,326\,696\,996\,318\,1 \times 10^{+12}$
18	16	$-3.958\,519\,466\,844\,432\,432\,722\,691\,784\,382\,239\,937\,383\,095\,314\,7 \times 10^{+13}$
20	0	$2.941\,412\,491\,004\,323\,318\,234\,070\,093\,506\,719\,255\,480\,372\,919\,4 \times 10^{+12}$
20	4	$5.079\,936\,332\,458\,166\,761\,279\,209\,566\,667\,515\,722\,321\,969\,772\,3 \times 10^{+11}$
20	8	$4.331\,937\,552\,580\,612\,828\,009\,012\,457\,415\,272\,217\,340\,902\,547\,5 \times 10^{+12}$
20	12	$4.778\,584\,572\,683\,966\,000\,847\,596\,132\,955\,755\,227\,539\,258\,877\,9 \times 10^{+13}$
20	16	$1.610\,388\,383\,673\,194\,996\,618\,067\,442\,771\,793\,943\,162\,737\,928\,3 \times 10^{+15}$
20	20	$5.010\,313\,960\,272\,320\,023\,745\,148\,415\,574\,132\,769\,616\,078\,384\,4 \times 10^{+17}$

Table A.4.: Non-zero high precision coefficients $\mathcal{L}_{lm}$ for cubic 3D periodic systems.

Bibliography

[1] G. Mittag-Leffler. The higher mathematics. *Nature*, 32(822):302–303, 1885.

[2] K.F. Sundman. Mémoire sur le problème des trois corps. *Acta Math.*, 36:105–179, 1913.

[3] Q.-D. Wang. The global solution of the N-body problem. *Celest. Mech. Dyn. Astr.*, 50:73–88, 1991.

[4] R.W. Hockney and J.W. Eastwood. *Computer Simulation Using Particles*. IOP, 1988.

[5] R. E. Mickens. Long-range interactions. *Found. Phys.*, 9:261–269, April 1979.

[6] Y. Yonetani. A severe artifact in simulation of liquid water using a long cut-off length: Appearance of a strange layer structure. *Chem. Phys. Lett.*, 406(1-3):49–53, 2005.

[7] R. P. Fedorenko. The speed of convergence of one iterative process. *USSR Comput. Math. Math. Phys.*, 4:227–235, 1964.

[8] N. S. Bakhvalov. On the convergence of a relaxation method with natural constraints on the elliptic operator. *USSR Comput. Math. Math. Phys.*, 6(5):101–135, 1966.

[9] A. Brandt. Multi-level adaptive solutions to boundary-value problems. *Math. Comput.*, 31(138):333–390, 1977.

[10] J. W. Eastwood, R. W. Hockney, and D. N. Lawrence. P3M3DP–the three-dimensional periodic particle-particle/ particle-mesh program. *Comput. Phys. Commun.*, 19(2):215–261, 1980.

[11] A. W. Appel. An efficient program for many-body simulation. *SIAM J. Sci. Stat. Comp.*, 6(1):85–103, 1985.

[12] J. Barnes and P. Hut. A hierarchical $\mathcal{O}(N \log N)$ force-calculation algorithm. *Nature*, 324:446–449, 1986.

[13] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.

[14] J. Dongarra and F. Sullivan. Guest editors' introduction: The top 10 algorithms. *Comput. Sci. Eng.*, 2:22–23, 2000.

[15] N. A. Gumerov and R. Duraiswami. *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions*. Elsevier, 2005.

[16] L. Greengard and J. Huang. A new version of the fast multipole method for screened Coulomb interactions in three dimensions. *J. Comput. Phys.*, 180(2):642–658, 2002.

[17] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comput. Phys.*, 73(2):325–348, 1987.

[18] N. Yarvin and V. Rokhlin. A generalized one-dimensional fast multipole method with application to filtering of spherical harmonics. *J. Comput. Phys.*, 147(2):594–609, 1998.

[19] N. Yarvin and V. Rokhlin. An improved fast multipole algorithm for potential fields on the line. *SIAM J. Numer. Anal.*, 36(2):629–666, 1999.

[20] L. Greengard and V. Rokhlin. A new version of the fast multipole method for the Laplace equation in three dimensions. *Acta Numer.*, 6:229–269, 1997.

[21] O. D. Kellogg. *Foundations of Potential Theory*. Springer, New York, 1967.

[22] P. R. Wallace. *Mathematical Analysis of Physical Problems*. Dover, New York, 1984.

[23] I. Gradshteyn and I. W. Ryzhik. *Tables of Integrals, Series and Products*. Academic Press, New York, 1965.

[24] M. E. Rose. *Elementary Theory of Angular Momentum*. Wiley, New York, 1957.

[25] E. P. Wigner. *Group theory and its application to the quantum mechanics of atomic spectra*. Academic Press, 1959.

[26] C. A. White and M. Head-Gordon. Derivation and efficient implementation of the fast multipole method. *J. Chem. Phys.*, 101:6593–6605, October 1994.

[27] M. J. Caola. Solid harmonics and their addition theorems. *J. Phys. A Math. Gen.*, 11(2):L23–L25, 1978.

[28] E. W. Hobson. *Spherical and Ellipsoidal Harmonics*. Cambridge University Press, New York, 1931.

[29] J. Choi. Notes on formal manipulations of double series. *Commun. Korean Math. Soc.*, 18(4):781–789, 2003.

[30] Wolfram Research. General indentities for analytic functions. `http://functions.wolfram.com/GeneralIdentities/12/`, March 2009.

[31] C. A. White and M. Head-Gordon. Rotating around the quartic angular momentum barrier in fast multipole method calculations. *J. Chem. Phys.*, 105(12):5061–5067, 1996.

[32] C. L. Berman. Grid-multipole calculations. *SIAM J. Sci. Comp.*, 16(5):1082–1091, 1995.

[33] W. D. Elliott and Jr. J. A. Board. Fast Fourier Transform accelerated fast multipole algorithm. *SIAM J. Sci. Comput.*, 17(2):398–415, 1996.

[34] H. Cheng, L. Greengard, and V. Rokhlin. A fast adaptive multipole algorithm in three dimensions. *J. Comput. Phys.*, 155(2):468–498, 1999.

[35] H. Dachsel, M. Hofmann, and G. Rünger. *Library Support for Parallel Sorting in Scientific Computations*, pages 695–704. Springer, 2007.

[36] C. A. White and M. Head-Gordon. Fractional tiers in fast multipole method calculations. *Chem. Phys. Lett.*, 257(5-6):647–650, 1996.

[37] D. E. Knuth. *The Art of Computer Programming.*, volume Vol. III: Sorting and Searching. Addison-Wesley Publishing Co., Menlo Park, 1973.

[38] R. A. Finkel and J. L. Bentley. Quad trees a data structure for retrieval on composite keys. *Acta Inform.*, 4(1):1–9, March 1974.

[39] H. Sagan. *Space-Filling Curves*. Springer-Verlag, 1994.

[40] M. F. Mokbel, W. G. Aref, and I. Kamel. Analysis of multi-dimensional space-filling curves. *GeoInformatica*, 7(3):179–209, 2003.

[41] S. Aluru, J. Gustafson, G. M. Prabhu, and F. E. Sevilgen. Distribution-independent hierarchical algorithms for the N-body problem. *J. Supercomput.*, 12(4):303–323, 1998.

[42] S. Aluru. Greengard's N-body algorithm is not order N. *SIAM J. Sci. Comput.*, 17(3):773–776, 1996.

[43] J. M. Pérez-Jordá and W. Yang. On the scaling of multipole methods for particle-particle interactions. *Chem. Phys. Lett.*, 282(1):71–78, 1998.

[44] J. H. Wilkinson. *Rounding Errors in Algebraic Processes*. Dover Publications, 1994.

[45] R. J. Buehler and J. O. Hirschfelder. Bipolar expansion of coulombic potentials. *Phys. Rev.*, 83(3):628–633, Aug 1951.

[46] B. C. Carlson and G. S. Rushbrooke. On the expansion of a Coulomb potential in spherical harmonics. *Math. Proc. Cambridge*, 46(04):626–633, 1950.

[47] D. Solvason and H. G. Petersen. Error estimates for the fast multipole method. *J. Stat. Phys.*, 86:391–420, 1997.

[48] G. J. Pringle. Numerical study of three-dimensional flow using fast parallel particle algorithms, 1994.

[49] W. Fong and E. Darve. The black-box fast multipole method. *J. Comput. Phys.*, 228(23):8712–8725, 2009.

[50] H. Dachsel. An error-controlled fast multipole method. *J. Chem. Phys.*, 132(11):119901, 2010.

[51] Wolfram Research Inc. Mathematica 7.0, 2008.

[52] M. E. Mura and N. C. Handy. Cuboidal basis functions. *Theor. Chem. Accounts. Theor. Comput. Model. Theor. Chim. Acta.*, 90(2):145–165, 1995.

[53] H. S. M. Coxeter. *Regular Polytopes*. Dover Publications, 1973.

[54] A. Leach. *Molecular Modelling: Principles and Applications (2nd Edition)*. Prentice Hall, March 2001.

[55] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of state calculations by fast computing machines. *J. Chem. Phys.*, 21(6):1087–1092, 1953.

[56] D. Borwein, J.M. Borwein, and K.F. Taylor. Convergence of lattice sums and Madelung's constant. *J. Math. Phys*, 26:2999–3009, 1995.

[57] P. P. Ewald. Die Berechnung optischer und elektrostatischer Gitterpotentiale. *Ann. Phys.*, 369(3):253–287, 1921.

[58] E. R. Smith. Electrostatic potentials in systems periodic in one, two, and three dimensions. *J. Chem. Phys.*, 128(17):174104, 2008.

[59] K. N. Kudin and G. E. Scuseria. Revisiting infinite lattice sums with the periodic fast multipole method. *J. Chem. Phys.*, 121(7):2886–2890, 2004.

[60] C. L. Berman and L. Greengard. A renormalization method for the evaluation of lattice sums. *J. Math. Phys.*, 35(11):6036–6048, 1994.

[61] M. Challacombe, C. White, and M. Head-Gordon. Periodic boundary conditions and the fast multipole method. *J. Chem. Phys.*, 107(23):10131–10140, 1997.

[62] A. Arnold, B. A. Mann, and Ch. Holm. Simulating charged systems with ESPResSo. In M. Ferrario, G. Ciccotti, and K. Binder, editors, *Computer Simulations in Condensed Matter: from Materials to Chemical Biology*, volume 703 of *Lecture Notes in Physics*, pages 193–222. Springer, 2006.

[63] V. Springel et al. Simulating the joint evolution of quasars, galaxies and their large-scale distribution. *Nature*, 435:629, 2005.

[64] R. E. Crandall and J. F. Delord. The potential within a crystal lattice. *J. Phys. A Math. Gen.*, 20(9):2279, 1987.

[65] R. E. Crandall. New representations for the madelung constant. *Exp. Math.*, 8(4):367–379, 1999.

[66] I. J. Zucker. Functional equations for poly-dimensional zeta functions and the evaluation of Madelung constants. *J. Phys. A Math. Gen.*, 9(4):499, 1976.

[67] M. B. Monagan, K. O. Geddes, K. M. Heal, G. Labahn, S. M. Vorkoetter, J. McCarron, and P. DeMarco. *Maple 10 Programming Guide*. Maplesoft, Waterloo ON, Canada, 2005.

[68] I. J. Zucker. Exact results for some lattice sums in 2, 4, 6 and 8 dimensions. *J. Phys. A Math. Gen.*, 7(13):1568, 1974.

[69] M. Abramowitz and I. A. Stegun. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Dover, New York, 1972.

[70] M. L. Glasser and I. J. Zucker. Lattice sums. In D. Henderson, editor, *Theoretical Chemistry : advances and perspectives.*, volume 5, pages 67–139. Academic Press. New York, 1980.

[71] IEEE Standard for Floating-Point Arithmetic. *IEEE Std 754-2008*, pages 1 –58, 29 2008.

[72] Juelich Supercomputing Centre. JUMP - Juelich Multi Processor. `http://www.fz-juelich.de/jsc/jump`, May 2010.

[73] S. Krumscheid. Benchmark of fast coulomb solvers for open and periodic boundary conditions. Technical Report FZJ-JSC-IB-2010-01, Juelich Supercomputing Centre, 2010.

[74] H.-J. Limbach, A. Arnold, B. A. Mann, and Ch. Holm. ESPResSo – an extensible simulation package for research on soft matter systems. *Comput. Phys. Commun.*, 174(9):704–727, May 2006.

[75] A. Arnold and Ch. Holm. MMM1D: A method for calculating electrostatic interactions in one-dimensional periodic geometries. *J. Chem. Phys.*, 123(14):144103, 2005.

[76] A. Arnold and Ch. Holm. MMM2D: A fast and accurate summation method for electrostatic interactions in 2d slab geometries. *Comput. Phys. Commun.*, 148:327–348, 2002.

[77] A. Arnold, J. Joannis, and Ch. Holm. Electrostatics in periodic slab geometries. I. *J. Chem. Phys.*, 117(6):2496–2502, 2002.

[78] V. Ballenegger, J. J. Cerdà, O. Lenz, and Ch. Holm. The optimal P3M algorithm for computing electrostatic energies in periodic systems. *J. Chem. Phys.*, 128(3):034109, 2008.

[79] A. Mikaberidze, U. Saalmann, and J. M. Rost. Laser-driven nanoplasmas in doped helium droplets: Local ignition and anisotropic growth. *Phys. Rev. Lett.*, 102(12):128102, 2009.

[80] U. Saalmann. Electron emission from laser-irradiated atomic clusters. *Laser Phys.*, 19(2):202–207, 2009.

[81] C. Gnodtke, U. Saalmann, and J. M. Rost. Ionization and charge migration through strong internal fields in clusters exposed to intense x-ray pulses. *Phys. Rev. A*, 79(4):041201, 2009.

[82] G. Bannasch and T. Pohl. Formation of atoms in strongly correlated, ultracold plasmas. Technical report, MPIPKS, 2008.

[83] I. Kabadshow. *The Fast Multipole Method - Alternative Gradient Algorithm and Parallelization*. Jül-4215. Forschungszentrum Jülich, 2006.

[84] I. Kabadshow and B. Lang. Latency-optimized parallelization of the FMM near-field computations. In *Comput. Sci. – ICCS 2007*. Springer, 2007.

[85] I. Chowdhury and V. Jandhyala. Single level multipole expansions and operators for potentials of the form $r^{-\lambda}$. *SIAM J. Sci. Comput.*, 26(3):930–943, 2005.

[86] J. P. Hansen. Statistical mechanics of dense ionized matter. I. Equilibrium properties of the classical one-component plasma. *Phys. Rev. A*, 8(6):3096–3109, 1973.

[87] J. J. Cerdà, V. Ballenegger, O. Lenz, and Ch. Holm. P3M algorithm for dipolar interactions. *J. Chem. Phys.*, 129:234104, 2008.

[88] BMBF Project 01 IH 08001 A-D. ScaFaCoS – Scalable Fast Coulomb Solver. `http://www.fz-juelich.de/jsc/scafacos`, May 2010.

1. **Three-dimensional modelling of soil-plant interactions: Consistent coupling of soil and plant root systems**
   by T. Schröder (2009), VIII, 72 pages
   ISBN: 978-3-89336-576-0
   URN: urn:nbn:de:0001-00505

2. **Large-Scale Simulations of Error-Prone Quantum Computation Devices**
   by D. B. Trieu (2009), VI, 173 pages
   ISBN: 978-3-89336-601-9
   URN: urn:nbn:de:0001-00552

3. **NIC Symposium 2010**
   Proceedings, 24 – 25 February 2010 | Jülich, Germany
   edited by G. Münster, D. Wolf, M. Kremer (2010), V, 395 pages
   ISBN: 978-3-89336-606-4
   URN: urn:nbn:de:0001-2010020108

4. **Timestamp Synchronization of Concurrent Events**
   by D. Becker (2010), XVIII, 116 pages
   ISBN: 978-3-89336-625-5
   URN: urn:nbn:de:0001-2010051916

5. **UNICORE Summit 2010**
   Proceedings, 18 – 19 May 2010 | Jülich, Germany
   edited by A. Streit, M. Romberg, D. Mallmann (2010), iv, 123 pages
   ISBN: 978-3-89336-661-3
   URN: urn:nbn:de:0001-2010082304

6. **Fast Methods for Long-Range Interactions in Complex Systems**
   Lecture Notes, Summer School, 6 – 10 September 2010**,** Jülich, Germany
   edited by P. Gibbon, T. Lippert, G. Sutmann (2011), ii, 167 pages
   ISBN: 978-3-89336-714-6
   URN: urn:nbn:de:0001-2011051907

7. **Generalized Algebraic Kernels and Multipole Expansions for Massively Parallel Vortex Particle Methods**
   by R. Speck (2011), iv, 125 pages
   ISBN: 978-3-89336-733-7
   URN: urn:nbn:de:0001-2011083003

8. **From Computational Biophysics to Systems Biology (CBSB11)**
   Proceedings, 20 - 22 July 2011 | Jülich, Germany
   edited by P. Carloni, U. H. E. Hansmann, T. Lippert, J. H. Meinke, S. Mohanty, W. Nadler, O. Zimmermann (2011), v, 255 pages
   ISBN: 978-3-89336-748-1
   URN: urn:nbn:de:0001-2011112819

9. **UNICORE Summit 2011**
   Proceedings, 7 - 8 July 2011 | Toruń, Poland
   edited by M. Romberg, P. Bała, R. Müller-Pfefferkorn, D. Mallmann (2011), iv, 150 pages
   ISBN: 978-3-89336-750-4
   URN: urn:nbn:de:0001-2011120103

10. **Hierarchical Methods for Dynamics in Complex Molecular Systems**
    Lecture Notes, IAS Winter School, 5 – 9 March 2012, Jülich, Germany
    edited by J. Grotendorst, G. Sutmann, G. Gompper, D. Marx (2012), ca. 500 pages
    ISBN: 978-3-89336-768-9
    URN: urn:nbn:de:0001-2012020208

11. **Periodic Boundary Conditions and the Error-Controlled Fast Multipole Method**
    by I. Kabadshow (2012), v, 126 pages
    ISBN: 978-3-89336-770-2
    URN: urn:nbn:de:0001-2012020810

The simulation of pairwise interactions in huge particle ensembles is a vital issue in scientific research. Especially the calculation of long-range interactions poses limitations to the system size, since these interactions scale quadratically with the number of particles N. Fast summation techniques like the Fast Multipole Method (FMM) can help to reduce the complexity to O(N).

This work extends the possible range of applications of the FMM to periodic systems in one, two and three dimensions with one unique approach. Together with a tight error control, this contribution enables the simulation of periodic particle systems for different applications without the need to know and tune the FMM specific parameters. The implemented error control scheme automatically optimizes the parameters to obtain an approximation for the minimal runtime for a given energy error bound.

This publication was written at the Jülich Supercomputing Centre (JSC) which is an integral part of the Institute for Advanced Simulation (IAS). The IAS combines the Jülich simulation sciences and the supercomputer facility in one organizational unit. It includes those parts of the scientific institutes at Forschungszentrum Jülich which use simulation on supercomputers as their main research methodology.

IAS Series
Volume 11
ISBN 978-3-89336-770-2

JÜLICH
FORSCHUNGSZENTRUM