Fast Ewald summation for electrostatic potentials with arbitrary periodicity

Davood Saffar Shamshirgar, Anna-Karin Tornberg

KTH Mathematics, Swedish e-Science Research Centre, 100 44 Stockholm, Sweden.

Abstract

A unified treatment for fast and spectrally accurate evaluation of electrostatic potentials subject to periodic boundary conditions in any or none of the three space dimensions is presented. Ewald decomposition is used to split the problem into a real space and a Fourier space part, and the FFT based Spectral Ewald (SE) method is used to accelerate the computation of the latter. A key component in the unified treatment is an FFT based solution technique for the free-space Poisson problem in three, two or one dimensions, depending on the number of non-periodic directions. The cost of calculations is furthermore reduced by employing an adaptive FFT for the doubly and singly periodic cases, allowing for different local upsampling rates. The SE method will always be most efficient for the triply periodic case as the cost for computing FFTs will be the smallest, whereas the computational cost for the rest of the algorithm is essentially independent of the periodicity. We show that the cost of removing periodic boundary conditions from one or two directions out of three will only marginally increase the total run time. Our comparisons also show that the computational cost of the SE method for the free-space case is typically about four times more expensive as compared to the triply periodic case.

The Gaussian window function previously used in the SE method, is here compared to an approximation of the Kaiser-Bessel window function, recently introduced in [2]. With a carefully tuned shape parameter that is selected based on an error estimate for this new window function, runtimes for the SE method can be further reduced.

Keywords: Fast Ewald summation, Fast Fourier transform, Arbitrary periodicity, Coulomb potentials, Adaptive FFT, Fourier integral, Spectral accuracy.

1. Introduction

The task of computing interactions in an N-body problem is the most demanding part of different numerical simulations such as electrostatics in molecular dynamics, gravitational fields in cosmological formation of galaxies, and potentials in Stokes flow simulations. Due to the long-range behavior of the involved kernels, these problems are computationally expensive and therefore, fast and accurate numerical algorithms are required to accelerate simulations. The Ewald technique [5] splits the interactions into a near field (computed in real space) and a far field (computed in Fourier space) contribution. There exist several methods that utilize this decomposition together with the Fast Fourier transform (FFT) in order to accelerate the calculation of the Fourier space sum [10, 3, 4, 12]. These methods belong to a family of

Particle-Mesh-Ewald (PME) methods which reduce the computational complexity from $\mathcal{O}(N^2)$ to $\mathcal{O}(N\log(N))$ with a prefactor depending on the required accuracy. The PME methods are known to be most efficient for fully periodic problems. As soon as a non-periodic direction exists, Fourier series are substituted with Fourier integrals that have to be resolved numerically before applying FFTs. During the last years, the Spectral Ewald method (SE) has been developed in order to provide a fast and spectrally accurate approach for evaluating electrostatics problems with arbitrary periodicities [12, 13, 1, 19]. The free-space and 1d-periodic versions of the method were developed recently and equipped with a novel technique proposed by Vico et al. which provides a tool for computing volume potentials using FFTs [21]. This extension makes it possible to unify the treatment of all modes in the singly periodic case and as a result, this case can be evaluated with a small extra cost as compared to the triply periodic case [19]. In [1], the free-space version of the SE method is used for accelerating the evaluation of free-space potentials of Stokes flow. It has been shown that the method is competitive with the Fast Multipole Method (FMM), which is most efficient for tackling problems with nonperiodic boundary conditions.

The SE method has so far been using a Gaussian window function to interpolate between point sources and a uniform mesh. In this paper, we replace the Gaussian by a new window function introduced recently by Barnett and Magland [2] to perform the interpolation. This window function is a modification of the original Kaiser-Bessel window function which retains desired properties and is substantially cheaper to evaluate. For both the Gaussian and the new window function, a shape parameter has to be set. We provide an error estimate useful for finding the optimal shape parameter of the modified Kaiser-Bessel window function and assess the accuracy of the estimate by means of numerical tests. We show that employing this window function instead of a Gaussian, the cost of evaluation using the SE method is reduced significantly.

This paper is organized as follows: In section 2, we provide Ewald summation formulas for different types of boundary conditions. In section 3, we construct the Spectral Ewald method for problems with different boundary conditions. Section 4 introduces different window functions that can be used in PME methods. Truncation and approximation errors together with approximation error estimates are introduced in section 5. The following section is dedicated to the parameter selection and precomputation step. We provide numerical results in section 7 and finalize with conclusions.

2. Ewald summation

The classical Ewald sum was developed for fast evaluation of potentials in ionic crystals and later the same technique was used for computing long-range interactions in molecular dynamics simulations and potentials in Stokes flow. The resulting formula relies on the Ewald decomposition introduced by Ewald in 1921 [5] for 3d-periodic problems. The Ewald sum in the 2d-periodic case, sometimes referred to as slab/slablike geometry, was derived by Grzybowski et al. [8] using lattice sums. The first derivation of the Ewald sum for the 1d-periodic case was given by Porto [17] using an integral representation of the Gamma function and the Poisson summation formula. The author left an integral in his expression, however, following [6], the closed form of the integral can be obtained. For alternative derivations of Ewald formulas with different periodicities, the reader may consult [20].

Consider a system of N point sources with charges q_n located at positions \mathbf{x}_n , $n = 1, 2, \dots, N$

in a cubic box $\Omega = [0, L)^3$. The objective is to calculate the following discrete sum

$$\varphi(\mathbf{x}_{m}) = \sum_{\mathbf{p} \in \mathcal{P}_{D}} \sum_{n=1}^{N} q_{n} G(\mathbf{x}_{m} - \mathbf{x}_{n} + \mathbf{p}), \tag{2.1}$$

where in the case of electrostatic calculations, $G(\mathbf{x}) = |\mathbf{x}|^{-1}$. Moreover, the prime in the summation denotes that the term with $\mathbf{n} = \mathbf{m}$ and $\mathbf{p} = 0$ is excluded from the sum. We also assume the system is charge neutral, i.e. $\sum_{\mathbf{n}} q_{\mathbf{n}} = 0$. This condition is necessary for the sum to converge in triply, doubly and singly periodic cases, however, we assume that it also holds for free-space systems. Moreover, \mathcal{P}_D with $D \in \{0, 1, 2, 3\}$ can be defined to impose periodicity,

Triply periodic: $\mathcal{P}_3 = \{(\alpha_1 L, \alpha_2 L, \alpha_3 L) : \alpha_i \in \mathbb{Z}\},$ Doubly periodic: $\mathcal{P}_2 = \{(\alpha_1 L, \alpha_2 L, 0) : \alpha_i \in \mathbb{Z}\},$ Singly periodic: $\mathcal{P}_1 = \{(\alpha_1 L, 0, 0) : \alpha_i \in \mathbb{Z}\},$

Free-space : $\mathcal{P}_0 = \{(0,0,0)\}.$

The sum in (2.1) is only conditionally convergent in the triply periodic case and therefore the order of summation has to be defined. The Ewald summation formula derived in [5] corresponds to a spherical order of summation.

The potential (2.1) is the solution to the problem

$$-\Delta\varphi(\mathbf{x}) = 4\pi f^{D\mathcal{P}}(\mathbf{x}), \quad f^{D\mathcal{P}}(\mathbf{x}) = \sum_{\mathbf{p}\in\mathcal{P}_D} \sum_{\mathbf{n}=1}^N q_{\mathbf{n}} \delta(\mathbf{x} - \mathbf{x}_{\mathbf{n}} + \mathbf{p}), \tag{2.2}$$

where δ is the Dirac delta function. By introducing a screening function, γ , f^{DP} is decomposed as,

$$f^{D\mathcal{P}} = f^{D\mathcal{P},R} + f^{D\mathcal{P},F}, \quad f^{D\mathcal{P},R} = f^{D\mathcal{P}} - (f^{D\mathcal{P}} * \gamma), \quad f^{D\mathcal{P},F} = (f^{D\mathcal{P}} * \gamma). \tag{2.3}$$

Now, $\varphi(\mathbf{x}_{\mathtt{m}})$ can be obtained by solving two Poisson equations with the right hand sides of $f^{D\mathcal{P},\mathtt{R}}$ and $f^{D\mathcal{P},\mathtt{F}}$. The solutions to these two problems are denoted here by $\varphi^{D\mathcal{P},\mathtt{R}}(\mathbf{x}_{\mathtt{m}})$ and $\varphi^{D\mathcal{P},\mathtt{F}}(\mathbf{x}_{\mathtt{m}})$ respectively. The total solution to the problem in (2.2) can then be written as

$$\varphi(\mathbf{x}_{m}) = \varphi^{D\mathcal{P},R}(\mathbf{x}_{m}) + \varphi^{D\mathcal{P},F}(\mathbf{x}_{m}) + \varphi^{\text{self}}_{m}.$$
(2.4)

To obtain the classical Ewald sum, the screening function and its Fourier transform, denoted as $\hat{\gamma}$, are defined as

$$\gamma(\mathbf{x},\xi) = \xi^3 \pi^{-3/2} e^{-\xi^2 |\mathbf{x}|^2}, \quad \hat{\gamma}(\mathbf{k},\xi) = e^{-|\mathbf{k}|^2/4\xi^2}. \tag{2.5}$$

In (2.5), ξ is the decomposition parameter which controls how fast the two terms $\varphi^{D\mathcal{P},R}(\mathbf{x}_m)$ and $\varphi^{D\mathcal{P},F}(\mathbf{x}_m)$ decay but does not change the final result.

The self contribution term in (2.4), $\varphi_{\mathtt{m}}^{\mathrm{self}}$, is a constant term which is independent of the periodicity. This term is added to the sum in order to exclude the interaction of charges with themselves which is contributed as a result of the decomposition. The term reads

$$\varphi_{\mathtt{m}}^{\mathrm{self}} = -\frac{2\xi}{\sqrt{\pi}}q_{\mathtt{m}}.$$

The real space sum $\varphi^{\mathcal{DP},R}(\mathbf{x}_m)$ can be written as

$$\varphi^{D\mathcal{P},R}(\mathbf{x}_{m}) = \sum_{\mathbf{p}\in\mathcal{P}_{D}}' \sum_{n=1}^{N} q_{n} \frac{\operatorname{erfc}(\xi|\mathbf{x}_{mnp}|)}{|\mathbf{x}_{mnp}|}, \quad m = 1, \dots, N,$$
(2.6)

where

$$\mathbf{x}_{\mathtt{mnp}} = \mathbf{x}_{\mathtt{mn}} + \mathbf{p} = \mathbf{x}_{\mathtt{m}} - \mathbf{x}_{\mathtt{n}} + \mathbf{p},$$

and as before $D \in \{0, 1, 2, 3\}$ represents free-space, 1d-, 2d- and 3d-periodic cases. This sum decays exponentially fast with $|\mathbf{x}_{mnp}|$ and is calculated by introducing a cut-off radius $r_c > 0$ and including only terms s.t. $|\mathbf{x}_{mnp}| < r_c$. In practice, a cell list is constructed for each target point \mathbf{x}_m . Taking into account that the domain is wrapped around periodically in periodic directions, the calculation is restricted to this list.

The $\varphi^{\mathcal{DP},F}(\mathbf{x}_m)$ term is smooth and therefore, its Fourier spectrum decays rapidly in Fourier space. For this term, periodicity has to be embedded into the formulation. To introduce the Fourier space contribution, we start by introducing some non standard notation in order to unify the descriptions for different periodicities.

Definition 1. Let $0 \le D \le 3$ be the number of periodic directions and $\mathbf{x} = [\mathbf{v}, \mathbf{w}] = (x, y, z)$ and $\mathbf{k} = [\mathbf{k}, \boldsymbol{\kappa}] = (k_1, k_2, k_3)$ where $\mathbf{w}, \boldsymbol{\kappa} \in \mathbb{R}^{3-D}$ and $\mathbf{v} \in \mathbb{R}^D$, $\mathbf{k} \in \mathcal{K}^D = \{\mathbf{k} : k_i \in \frac{2\pi}{L_i}\mathbb{Z}, i = 1, ..., D\}$. For D = 3, \mathbf{w} and $\boldsymbol{\kappa}$ are not defined and $\mathbf{k} = \mathbf{k}$, $\mathbf{x} = \mathbf{v}$. For D = 0, \mathbf{v} and \mathbf{k} are not defined and $\mathbf{k} = \boldsymbol{\kappa}$, $\mathbf{x} = \mathbf{w}$. Also $|\mathbf{k}| = k$ and $|\boldsymbol{\kappa}| = \kappa$. Let $f([\mathbf{v}, \mathbf{w}])$ be a function that is periodic in \mathbf{v} with the Fourier transform $\hat{f}([\mathbf{k}, \boldsymbol{\kappa}])$ and consider the following notation,

$$\mathcal{L}[\hat{f}(\mathbf{k})] = \mathcal{L}[\hat{f}([\mathbf{k}, \boldsymbol{\kappa}])] := \begin{cases} \frac{1}{L^{D}} \sum_{\mathbf{k} \in \mathcal{K}^{D}} \hat{f}(\mathbf{k}), & D = 3, \\ \frac{1}{(2\pi)^{3-D} L^{D}} \sum_{\mathbf{k} \in \mathcal{K}^{D}} \int_{\mathbb{R}^{3-D}} \hat{f}([\mathbf{k}, \boldsymbol{\kappa}]) d\boldsymbol{\kappa}, & D \in \{1, 2\}, \\ \frac{1}{(2\pi)^{3-D}} \int_{\mathbb{R}^{3-D}} \hat{f}(\mathbf{k}) d\mathbf{k}, & D = 0. \end{cases}$$

$$(2.7)$$

Then f and \hat{f} are related as

$$f([\mathbf{v}, \mathbf{w}]) = L^D \mathcal{L}[\hat{f}([\mathbf{k}, \boldsymbol{\kappa}]) e^{i\mathbf{k}\cdot\mathbf{v}} e^{i\boldsymbol{\kappa}\cdot\mathbf{w}}]. \tag{2.8}$$

Using this notation, the k-space part of the Ewald sum $\varphi^{D\mathcal{P},F}(\mathbf{x}_{\mathtt{m}})$ reads

$$\varphi^{D\mathcal{P},F}(\mathbf{x}_{m}) = 4\pi \sum_{n=1}^{N} q_{n} \mathcal{L} \left[\frac{e^{-(\mathbf{k}^{2} + \kappa^{2})/4\xi^{2}}}{\mathbf{k}^{2} + \kappa^{2}} e^{i\mathbf{k}\cdot(\mathbf{v}_{m} - \mathbf{v}_{n})} e^{i\mathbf{\kappa}\cdot(\mathbf{w}_{m} - \mathbf{w}_{n})} \right], \tag{2.9}$$

For D=3, the $\mathbf{k}=0$ term depends on the order of summation and vanishes for the case of infinite dielectric constant [12]. For D=0, the operator \mathcal{L} only includes Fourier integrals

defined for all $\kappa \in \mathbb{R}^3$ and the closed form of the integral is nothing but the complement of the real space sum minus the self term.

For $D \in \{1, 2\}$, the Fourier integrals are defined for discrete modes $\mathbf{k} \in \mathcal{K}^D$. For $\mathbf{k} \neq 0$ these integrals can be evaluated analytically. We have

$$\varphi^{2\mathcal{P},F,k\neq 0}(\mathbf{x}_{m}) = \frac{\pi}{L^{2}} \sum_{\mathbf{n}=1}^{N} \sum_{\mathbf{k}\neq 0} q_{\mathbf{n}} \frac{e^{\mathrm{i}\mathbf{k}\cdot\mathbf{v}_{mn}}}{k} \left[e^{kz_{mn}} \operatorname{erfc}\left(\frac{k}{2\xi} + \xi z_{mn}\right) e^{-kz_{mn}} \operatorname{erfc}\left(\frac{k}{2\xi} - \xi z_{mn}\right) \right], \quad (2.10)$$

$$\varphi^{1\mathcal{P},F,k\neq 0}(\mathbf{x}_{m}) = \frac{1}{L} \sum_{n=1}^{N} \sum_{k_{1}\neq 0} q_{n} e^{ik_{1}x_{mn}} \mathbf{K}_{0}(k_{1}^{2}/4\xi^{2}, |\mathbf{w}_{mn}|^{2}\xi^{2}), \tag{2.11}$$

in which we used the fact that for D=2, $\mathbf{w}_{mn}=z_{mn}$ and for D=1, $\mathbf{v}_{mn}=x_{mn}$ and $\mathbf{k}=k_1$. Also, $\mathbf{K}_0(\cdot,\cdot)$ is the incomplete modified Bessel function of the second kind defined as

$$\mathbf{K}_0(a,b) = \int_1^\infty \frac{\mathrm{d}t}{t} e^{-at-b/t}.$$

Note that, as it is done in [15], it is possible to construct a fast method for the sums defined in (2.10) and (2.11). But using the integral representation of the sums (2.9), we are able to construct a fast method that has a minimal deviation from the treatment of the triply periodic SE method [12] while incurring only a small additional cost.

For $D \in \{1, 2\}$ and $\mathbf{k} = 0$, the Fourier integrals in (2.9) are singular but there are closed form solutions

$$\varphi^{2\mathcal{P},F,k=0}(\mathbf{x}_{m}) = -\frac{2\sqrt{\pi}}{L^{2}} \sum_{n=1}^{N} q_{n} \left[e^{-\xi^{2} z_{mn}^{2}} / \xi + \sqrt{\pi} z_{mn} \operatorname{erf}(\xi z_{mn}) \right], \qquad (2.12)$$

$$\varphi^{1\mathcal{P},F,k=0}(\mathbf{x}_{m}) = -\frac{1}{L} \sum_{\substack{n=1\\n\neq m}}^{N} q_{n} \{ \gamma + \log(\xi^{2} |\mathbf{w}_{mn}|^{2}) + \mathcal{E}_{1}(\xi^{2} |\mathbf{w}_{mn}|^{2}) \},$$
(2.13)

where $E_1(\cdot) = \mathbf{K}_0(\cdot,0)$ and $\gamma = 0.5772156649...$ is the Euler-Mascheroni constant. The term $\varphi^{2\mathcal{P},F,k=0}$, cf. (2.12), is a one dimensional sum and in [13] it is computed via Chebyshev interpolations. This approach is much more expensive if it is to be used for the two dimensional sum $\varphi^{1\mathcal{P},F,k=0}$. In fact, using Chebyshev interpolations in this case, the cost of the zero mode term (2.13) is comparable with the cost of calculating the rest of the Fourier modes, cf. (2.11). Instead, we note that for $\mathbf{k} = 0$, (2.12) and (2.13) are solutions to (3-D)-dimensional free-space Poisson problems. Using the idea in [21] the corresponding forms in (2.9) can be replaced by non-singular expressions amenable to numerical integration. This approach has already been used in [19, 1] for designing the Spectral Ewald method for the 1d-periodic and free-space cases. In this paper, the same technique is used for the zero mode of the 2d-periodic case as well. In section 3.1 we review the free-space Poisson solver in detail.

3. The Spectral Ewald method

Here, we introduce a unified fast method to accelerate the computation of the Fourier space part of the electrostatic potential.

The Spectral Ewald method (SE) was first introduced by Lindbo and Tornberg [12] for the 3d-periodic case and was extended by the same authors for the 2d-periodic case [13]. The triply periodic SE method follows the same steps as the other PME methods. First, a uniform grid is introduced and irregular point sources are distributed on the grid using an interpolation (window) function (gridding step). In the case of the SE method, Gaussians were previously used as interpolation functions. Having point sources distributed on the uniform grid, in the next step, an FFT is applied to compute the Fourier transform of the gridded function. In Fourier space, the result is then scaled with a modified Green's function (scaling step) and an IFFT is employed to take the result back to real space. Finally, the Fourier space part of the potential (2.9) is evaluated at target points using an interpolation with the same window function (gathering step).

The choice of window function influences the accuracy and runtime of the resulting method. Moreover, and as a result of this selection, the modified Green's function has to be adjusted accordingly.

3.1. Free-space Poisson solver

In this section we briefly introduce the main idea of the free-space Poisson solver in three dimensions and refer the reader to the original reference [21] for more details, to [1] for an application in three dimensions and to [19] for an application in two dimensions.

Consider the following Poisson equation defined on $\Omega = [0, L]^3$ with free-space boundary conditions $(\varphi(\mathbf{x}) \to 0 \text{ as } |\mathbf{x}| \to \infty)$

$$-\Delta\varphi(\mathbf{x}) = f(\mathbf{x}).$$

The solution to this problem can be written in real space as a convolution with the right hand side $f(\mathbf{x})$ or in Fourier space as

$$\varphi(\mathbf{x}) = \int_{\mathbb{R}^3} G(\mathbf{x} - \mathbf{y}) f(\mathbf{y}) d\mathbf{y} = \frac{1}{(2\pi)^3} \int_{\mathbb{R}^3} \widehat{G}(\mathbf{k}) \widehat{f}(\mathbf{k}) e^{i\mathbf{k} \cdot \mathbf{x}} d\mathbf{k}, \tag{3.1}$$

where $G(\mathbf{x}) = 1/(4\pi|\mathbf{x}|)$. Now let $f(\mathbf{x})$ be a compactly supported function defined on the extended domain $\tilde{\Omega} = [0, \tilde{L}]^3$ and define $R = |\tilde{\Omega}| = \sqrt{3}\tilde{L}$. Moreover, assume that we seek the solution of $\varphi(\mathbf{x})$ in $\tilde{\Omega}$. We can now replace $G(\mathbf{x}) = G(|\mathbf{x}|) = G(r)$ in (3.1) with a truncated version

$$G_{\scriptscriptstyle R}(r) = G(r) \cdot \operatorname{rect}\left(\frac{r}{2R}\right),$$

where

$$rect(x) = \begin{cases} 1, & |x| \le 1/2, \\ 0, & |x| > 1/2. \end{cases}$$

Then, using the fact that $G_R(r)$ is radially symmetric, its Fourier transform can be computed as

$$\widehat{G}_{\scriptscriptstyle R}(k) = 2 \left(\frac{\sin(Rk/2)}{k} \right)^2,$$

and in the limit we have

$$\widehat{G}_{R}(0) = \lim_{k \to 0} \widehat{G}_{R}(k) = \frac{R^{2}}{2}.$$

Using the technique explained above, we can derive corresponding Fourier transforms also for one and two dimensional free-space Poisson equations.

Dropping the subscript R and reminding that $\widehat{G}(\mathbf{k}) = 1/|\mathbf{k}|^2$ for non-singular cases, $\widehat{G}(\mathbf{k})$ is defined as follows:

For D=3 (periodic in all directions), $\mathbf{k}=\mathbf{k}=(k_1,k_2,k_3)$ and $\boldsymbol{\kappa}$ is not defined, and

$$\widehat{G}(\mathbf{k}) = \begin{cases} \frac{1}{|\mathbf{k}|^2} & \mathbf{k} \neq \mathbf{0}, \\ 0 & \mathbf{k} = \mathbf{0}. \end{cases}$$
(3.2)

For D=2 (periodic in the x and y directions and free in the z direction), $\mathbf{k}=(k_1,k_2)$ and $\mathbf{\kappa}=k_3$, and

$$\widehat{G}(\mathbf{k}) = \widehat{G}([\mathbf{k}, \boldsymbol{\kappa}]) = \begin{cases} \frac{1}{|\mathbf{k}|^2} & \mathbf{k} \neq \mathbf{0}, \\ \frac{(R|\boldsymbol{\kappa}|\sin(R|\boldsymbol{\kappa}|) + \cos(R|\boldsymbol{\kappa}|) - 1)}{|\boldsymbol{\kappa}|^2} & \mathbf{k} = \mathbf{0}, \boldsymbol{\kappa} \neq \mathbf{0}, \\ \frac{R^2}{2} & \mathbf{k} = \mathbf{0}. \end{cases}$$
(3.3)

For D=1 (periodic in the x direction and free in the y and z directions), $\mathbf{k}=k_1$ and $\mathbf{\kappa}=(k_2,k_3)$, and

$$\widehat{G}(\mathbf{k}) = \widehat{G}([\mathbf{k}, \boldsymbol{\kappa}]) = \begin{cases} \frac{1}{|\mathbf{k}|^2} & \mathbf{k} \neq \mathbf{0}, \\ \frac{1 - J_0(R|\boldsymbol{\kappa}|)}{|\boldsymbol{\kappa}|^2} - \frac{R\log(R)J_1(R|\boldsymbol{\kappa}|)}{|\boldsymbol{\kappa}|} & \mathbf{k} = \mathbf{0}, \boldsymbol{\kappa} \neq \mathbf{0}, \\ \frac{R^2}{4}(1 - 2\log(R)) & \mathbf{k} = \mathbf{0}. \end{cases}$$
(3.4)

And finally for D=0 (free in all directions), $\kappa=\mathbf{k}=(k_1,k_2,k_3)$ and k is not defined, and

$$\widehat{G}(\mathbf{k}) = \begin{cases} 2\left(\frac{\sin(R|\mathbf{k}|/2)}{|\mathbf{k}|}\right)^2 & \mathbf{k} \neq \mathbf{0}, \\ \frac{R^2}{2} & \mathbf{k} = \mathbf{0}. \end{cases}$$
(3.5)

In the derivation above, we have assumed that the right hand side to be compactly supported and R to be sufficiently large. In (2.3), however, the right hand side is a superposition of screening functions, i.e. Gaussians, and hence does not have compact support. Nonetheless, in practice, it decays rapidly outside of $\Omega = [0, L)^3$ and the extended domain $\tilde{\Omega}$ can be selected such that the magnitude is arbitrary small. Having this in mind, inserting \hat{G} into (2.9), we have

$$\widetilde{\varphi}^{D\mathcal{P},F}(\mathbf{x}_{m}) = 4\pi \sum_{n=1}^{N} q_{n} \mathcal{L} \left[e^{-(\mathbf{k}^{2} + \kappa^{2})/4\xi^{2}} \widehat{G}([\mathbf{k}, \kappa]) e^{i\mathbf{k}\cdot(\mathbf{v}_{m} - \mathbf{v}_{n})} e^{i\mathbf{k}\cdot(\mathbf{w}_{m} - \mathbf{w}_{n})} \right], \tag{3.6}$$

in which $\widetilde{\varphi}^{3\mathcal{P},F}(\mathbf{x}_m) = \varphi^{3\mathcal{P},F}(\mathbf{x}_m)$ and $\widetilde{\varphi}^{D\mathcal{P},F}(\mathbf{x}_m) \approx \varphi^{D\mathcal{P},F}(\mathbf{x}_m)$ for $D = \{0,1,2\}$. For the triply periodic case (3.6) is just another representation of (2.9). For the doubly and singly periodic cases, it introduces an approximation to the unified representation of the zero Fourier modes (2.12) and (2.13) with (2.9). Finally for the free-space case, it is an approximate solution to the Fourier space sum given in the form of a singular triple Fourier integral (2.9).

3.2. Spreading, scaling and gathering

So far, we have presented a unified formula for the representation of the Fourier space sum with different periodicities. Here, we introduce the procedure for incorporating window functions in the spreading, scaling and gathering steps. Later in section 4, we review four relevant window functions that are used in different Ewald methods.

Assume that $w(\mathbf{x}, \xi)$ is a window function with the Fourier transform $\widehat{w}(\mathbf{k}, \xi)$ and consider the following trivial identity

$$1 \equiv \widehat{\mathbf{w}}(\mathbf{k}, \xi)\widehat{\mathbf{w}}^{-2}(\mathbf{k}, \xi)\widehat{\mathbf{w}}(\mathbf{k}, \xi). \tag{3.7}$$

The Fourier transform of the window function $\widehat{\mathbf{w}}$ is used in the gridding and gathering steps, and $\widehat{\mathbf{w}}^{-2}$ is employed in the scaling step. Now consider again the Fourier space sum in (3.6). Using $\widehat{\mathbf{w}}$ we define

$$\widehat{H}(\mathbf{k}) := \sum_{n=1}^{N} q_n \widehat{\mathbf{w}}(\mathbf{k}, \xi) e^{i\mathbf{k} \cdot \mathbf{x}_n}, \tag{3.8}$$

which is the Fourier transform of

$$H(\mathbf{x}) = \sum_{n=1}^{N} q_n \mathbf{w}(\mathbf{x} - \mathbf{x}_n, \xi)_*. \tag{3.9}$$

Here, $(\cdot)_*$ denotes that the periodicity is implied in the periodic directions. We furthermore define

$$\widehat{\widetilde{H}}(\mathbf{k}) := e^{-\mathbf{k}^2/4\xi^2} \widehat{\mathbf{w}}^{-2}(\mathbf{k}, \xi) \widehat{G}(\mathbf{k}) \widehat{H}(\mathbf{k}), \tag{3.10}$$

in which \widehat{G} is the modified Green's function defined in section 3 in equations (3.2)-(3.5). We can write

$$\widetilde{\varphi}^{D\mathcal{P},F}(\mathbf{x}) = 4\pi \mathcal{L} \left[\widehat{\mathbf{w}}(\mathbf{k},\xi) \widehat{\widetilde{H}}(-\mathbf{k}) e^{i\mathbf{k}\cdot\mathbf{v}} e^{i\mathbf{\kappa}\cdot\mathbf{w}} \right],$$
(3.11)

Applying the Plancherel and convolution theorems on (3.11) we obtain the Fourier space part of the Ewald sum for a target point \mathbf{x}_m in an integral form

$$\widetilde{\varphi}^{D\mathcal{P},F}(\mathbf{x}_{m}) = 4\pi \int_{\mathbb{R}^{3-D}} \int_{[0,L)^{D}} \widetilde{H}(\mathbf{v}, \mathbf{w}) \mathsf{w}(\mathbf{x}_{m} - \mathbf{x}, \xi)_{*} d\mathbf{v} d\mathbf{w}. \tag{3.12}$$

The integral in (3.12) can be discretized and computed using the trapezoidal rule. We shall note here that if w is smooth and have compact support, the integral can be computed with spectral accuracy.

In the above procedure, we obtain \widehat{H} from H and \widetilde{H} from \widetilde{H} via Fourier transforms. For the Fourier space Ewald sum in 2d- and 1d-periodic cases these transforms are mixed: a discrete Fourier transform in each periodic direction and an approximation to the continuous Fourier integral in each non periodic direction. We shall return to this discussion later in section 3.4.

3.3. Discretization

We start by introducing a uniform grid of size M on $[0,L)^3$ and define h=L/M. Then, charges are assigned to the uniform grid using a suitable window function (gridding step, cf. (3.9)). In non-periodic directions, the grid needs to be extended in order to accommodate for the support of window functions. Let P, where $P \leq M$, be the number of points in the support of a window function in each direction. Hence, in non-periodic directions we have L = L + Ph and we require that the grid size to be increased as M = M + P such that the grid spacing remains the same, i.e. $h = L/M = \tilde{L}/\tilde{M}$. Before applying FFTs, we note that Fourier integrals in non periodic directions have to be discretized on a finer grid. The simplest way to handle this is to define a global upsampling factor s_g and extend the domain to $s_a\tilde{L}$. Here, upsampling means zero padding in real space which leads to a denser sampling of modes in Fourier space. As a result of upsampling, the grid is extended to $s_q M$ grid points and again h remains unchanged. Now we can apply FFTs on a grid of size $M^D(s_q\tilde{M})^{3-D}$ to compute Fourier coefficients for the k-space vectors $k = 2\pi n/L$, $n = -M/2, \ldots, M/2 - 1$ in periodic directions and $\kappa = 2\pi n/(s_q \tilde{L}), n = -s_q \tilde{M}/2, \dots, s_q \tilde{M}/2 - 1$ in non periodic directions. Clearly, for smaller D, the grid is larger and the cost of calculating FFTs is bigger. In the next section, we introduce an adaptive Fourier transform AFT to apply different upsampling factors locally for D=1,2. For D=3 there is no need for upsampling and for D=0, part of the cost can be hidden in a precomputation step instead. We shall then observe that applying these techniques, the cost of calculation reduces significantly. The scaling step, cf. (3.10), can be performed on the extended upsampled grid with $M^D(s_g\tilde{M})^{3-D}$ grid points. The inverse AFT, AIFT, is then applied and the result is restricted to $\tilde{M}^D \tilde{M}^{3-D}$ grid points. Finally, the gathering step (evaluating (3.12) with the trapezoidal rule) is performed to obtain potentials at target points.

3.4. Fourier integrals and adaptive FFT

In our formulation, two types of Fourier (and inverse Fourier) integrals are present. One for the zero and one for non zero Fourier modes. Both integral types need to be discretized using the trapezoidal rule and evaluated efficiently using FFTs. Each type however, requires a different upsampling rate. Henceforth, we use s_0 to denote the upsampling rate of Fourier integrals for zero Fourier modes and s for the upsampling rate of non zero Fourier modes. It has been shown that for free-space Poisson problems in d dimensions, $s_0 \ge 1 + \sqrt{d}$ is required to accurately compute the aperiodic convolution and also to account for the oscillatory behavior of Fourier integrals [1]. We have also shown that no upsampling is required to compute Fourier integrals but we require up to s=4 upsampling rate to resolve inverse Fourier integrals to machine accuracy [19]. With a global upsampling rate of 4, the total grid size for D=2 is $4M^2 \times \tilde{M}$ and for D=1 is $16M \times \tilde{M}^2$ which is clearly not desirable. An analysis of a similar inverse Fourier integral in one dimension, however, shows that for large values of $|\mathbf{k}|$ (discrete modes in periodic directions) the integrand decays to zero and therefore there is no need for upsampling to resolve the inverse Fourier integral [19]. Using this observation, we developed an adaptive Fourier transform (AFT) which applies different upsampling rates on different modes. In figure 1 we present a schematic representation of this adaptive Fourier transform in two dimensions. The procedure is as follows. First, let n be a positive integer such that $n \ll k_{\infty} = M/2$ and define the following sets

$$\mathbb{I} := \{ \mathbf{k} : 0 < |k_i| \le \frac{2\pi}{L} \mathsf{n}, i \le D \}, \tag{3.13}$$

and

$$\mathbb{J} := \{ \mathbf{k} : \frac{2\pi}{L} \mathsf{n} < |k_i| \le \frac{2\pi}{L} k_{\infty}, i \le D \}.$$
 (3.14)

Given upsampling rates s_0 and s, the AFT computes the Fourier transform $\mathbf{w} \to \boldsymbol{\kappa}$ for $\mathbf{k} = 0$ with an oversampling factor s_0 , for $\mathbf{k} \in \mathbb{I}$ with an oversampling factor s_0 , and for $\mathbf{k} \in \mathbb{I}$ with no oversampling. Therefore, the adaptive upsampling factor can be defined as

$$s_f = \begin{cases} s_0, & |k_i| = 0, \\ s, & 1 \le |k_i| \le \frac{2\pi}{L} \mathsf{n}, \\ 1, & |k_i| > \frac{2\pi}{L} \mathsf{n}. \end{cases}$$
 (3.15)

Using these upsampling rates and by reminding the notations introduced in definition 1, we present the AFT/AIFT algorithms. These algorithms are the generalized versions of those introduced in [19].

Algorithm 1 Adaptive Fourier transform - AFT for D = 1, 2

Input: Grid representation of sources H, upsampling factors s_0 , s, and \mathbb{I} , \mathbb{J} sets, cf. (3.13) and (3.14).

- 1: Apply an FFT on H in the \mathbf{v} directions to compute $\widehat{H}(\mathbf{k}, \mathbf{w})$.
- 2: Pad $\widehat{H}(\mathbf{0}, \mathbf{w})$ with zeros in the \mathbf{w} directions with upsampling factor s_0 and apply an FFT to compute $\widehat{H}(\mathbf{0}, \boldsymbol{\kappa})$.
- 3: Pad $\widehat{H}(\mathbb{I}, \mathbf{w})$ with zeros in the \mathbf{w} directions with upsampling factor s and apply an FFT on $\mathbf{k} \in \mathbb{I}$ to compute $\widehat{H}(\mathbb{I}, \mathbf{\kappa})$.
- 4: Apply an FFT on $\widehat{H}(\mathbb{J}, \mathbf{w})$ with no upsampling to compute $\widehat{H}(\mathbb{J}, \boldsymbol{\kappa})$.

Output: Adaptive Fourier transform of H, H.

Note that if $n = k_{\infty}$, all non zero Fourier modes are upsampled and therefore, steps 3 and 4 can be merged. If we also have $s = s_0$, the AFT is the same as a plain three dimensional FFT.

Now, assume that as a result of applying AFT algorithm, $\widehat{H}(\mathbf{0}, \kappa)$, $\widehat{H}(\mathbb{I}, \kappa)$ and $\widehat{H}(\mathbb{J}, \kappa)$ are available. To compute H from \widehat{H} in adaptive inverse FFT algorithm (AIFT), we can easily reverse the steps in AFT algorithm.

Algorithm 2 Adaptive inverse Fourier transform - AIFT for D = 1, 2

Input: $\hat{H}(\mathbf{0}, \kappa)$, $\hat{H}(\mathbb{I}, \kappa)$ and $\hat{H}(\mathbb{J}, \kappa)$, grid size M and \mathbb{I}, \mathbb{J} sets, cf. (3.13) and (3.14).

- 1: Apply an IFFT on $\widehat{H}(\mathbb{J}, \kappa)$ in the w directions to compute $\widehat{H}(\mathbb{J}, \mathbf{w})$.
- 2: Apply an IFFT on $\widehat{H}(\mathbb{I}, \kappa)$ in the **w** directions and restrict the solution to M grid points in each direction to compute $\widehat{H}(\mathbb{I}, \mathbf{w})$.
- 3: Apply an IFFT on $\widehat{H}(\mathbf{0}, \kappa)$ in the **w** directions and restrict the solution to M grid points in each direction to compute $\widehat{H}(\mathbf{0}, \kappa)$.
- 4: Merge $\hat{H}(\mathbb{I}, \mathbf{w})$, $\hat{H}(\mathbb{J}, \mathbf{w})$ and $\hat{H}(\mathbf{0}, \mathbf{w})$ to construct $\hat{H}(\mathbf{k}, \mathbf{w})$.
- 5: Apply an IFFT on $H(\mathbf{k}, \mathbf{w})$ in the \mathbf{v} directions to compute $H(\mathbf{v}, \mathbf{w})$.

Output: Adaptive inverse Fourier transform of \widehat{H} , H.

Once again, if $n = k_{\infty}$ and $s = s_0$, all steps can be merged into a three dimensional inverse FFT followed by restricting the result to M^3 grid points. It also worth mentioning that for D = 3, 0, the AFT/AIFT is simply a three dimensional FFT/IFFT on a grid of size M^3 and $(s_0\tilde{M})^3$ respectively.

In Algorithm 3, we present the Spectral Ewald method to compute the approximate Fourier space part of the electrostatic potential in (3.6) with different periodicities.

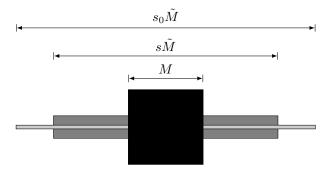


Figure 1: Schematic representation of mixed Fourier Transforms in two dimensions.

Algorithm 3 Spectral Ewald method - k-space algorithm

Input: Charge locations $\mathbf{x}_n \in [0, L)^3$ and charges q_n , n = 1, ..., N, splitting parameter ξ , grid size M, oversampling factors s_0 , s, maximum upsampled Fourier modes n, number of points in the support of window functions P, number of periodic directions D.

- 1: Set h = L/M, $\tilde{L} = L + Ph$, $\tilde{M} = M + P$, $R = \sqrt{3 D}\tilde{L}$.
- 2: Introduce a uniform grid on $[0, L)^D \times [0, \tilde{L})^{3-D}$ with $M^D \times \tilde{M}^{3-D}$ points. Evaluate $H([\mathbf{v}, \mathbf{w}])$ on this grid according to (3.9).
- 3: Apply an AFT with s_0 , s and n to compute $\widehat{H}([\mathbf{k}, \kappa])$.
- 4: Scale as in (3.10) to obtain \widetilde{H} .
- 5: Apply an AIFT to compute \widetilde{H} .
- 6: Compute the integral in (3.12) at target points (same as source points) using the trapezoidal rule.

Output: Approximation to the Fourier space part of the potentials $\varphi^{D\mathcal{P},\hat{F}}(\mathbf{x}_m)$, $m=1,\ldots,N$.

In molecular dynamics simulations, one is usually interested in evaluating the force and energy along with the potential. The energy can be obtained simply by using

$$E = \sum_{\mathtt{m}=1}^{N} q_{\mathtt{m}} \varphi^{D\mathcal{P}, \mathrm{F}}(\mathbf{x}_{\mathtt{m}}).$$

To compute the force, the potential has to be differentiated. In order to preserve spectral accuracy, one can differentiate (3.12) with respect to \mathbf{x}_m . Therefore, the same approach as in Algorithm 3 can be used to calculate the force. However, the result in step 5, is a three-dimensional vector and therefore, the integration in this step has to be performed three times, [19].

4. Window functions

Here, we review some of the most relevant window functions that appear in the electrostatic calculations and specifically in PME methods. Moreover, we review the most recent window function introduced by Barnett *et al.* [2]. For a complete survey on the classical window functions, the reader is directed to [9].

The following window functions are presented in one dimension. In three dimensions, the corresponding window function $w(\mathbf{x})$ can be obtained by a tensor product as

$$w(\mathbf{x}) = w(x)w(y)w(z).$$

Gaussian window. Gaussians have several important properties. First and foremost, Fourier transforms of Gaussians are available and are also Gaussians. Moreover, they are smooth and easy to implement and decay rapidly in Fourier space. Nonetheless, they do not have compact support and therefore have to be truncated in practice. As noted earlier, in the SE method [1, 12, 13, 19], Gaussians are used as window functions. Introducing a free parameter $\eta > 0$, the window and its Fourier transform are defined as

$$g(x,\xi) = \left(\frac{2\xi^2}{\pi\eta}\right)^{1/2} e^{-2\xi^2 x^2/\eta}, \quad \hat{g}(k,\xi) = e^{-\eta k^2/8\xi^2}. \tag{4.1}$$

Here, $\eta = (2w\xi/m)^2$ such that m controls the truncation level of the Gaussian as it is truncated at |x| = w, see also section 5.2. Note that in higher dimensions, the window is truncated outside of a cube.

Cardinal B-spline window. Another type of window functions that are used in FFT-based methods are cardinal B-splines [4]. The B-spline of order 2 is defined as

$$M_2(x) = \begin{cases} 1 - |x - 1|, & 0 \le x \le 2, \\ 0, & \text{otherwise,} \end{cases}$$

and for p > 2 is defined recursively as

$$M_p(x) = \frac{x}{p-1} M_{p-1}(x) + \frac{p-x}{p-1} M_{p-1}(x-1).$$
(4.2)

The window has finite support and is easy and fast to implement. Moreover, its Fourier transform is available analytically. B-splines have polynomial degree of smoothness and consequently, whenever they are used in an FFT-based method, in order to reduce approximation errors and to achieve high accuracies, the FFT grid size must be increased significantly.

Kaiser-Bessel window. The *Kaiser-Bessel* function can also serve as a window kernel. This function is defined as

$$K(x) = \begin{cases} \frac{I_0 \left(\beta \sqrt{1 - \left(\frac{x}{w}\right)^2}\right)}{I_0(\beta)}, & -w \le x \le w, \\ 0, & \text{otherwise,} \end{cases}$$
(4.3)

in which $I_0(\cdot)$ is the zeroth-order modified Bessel function of the first kind and $\beta > 0$ is denoted as the shape parameter. The Fourier transform of the window is also available in closed form

$$\widehat{K}(k) = \frac{\sinh(\sqrt{\beta^2 - k^2 w^2})}{I_0(\beta)\sqrt{\beta^2 - k^2 w^2}}.$$
(4.4)

This Kaiser-Bessel window is an approximation to a family of prolate-spheroidal wave functions of order zero. It has been shown that the prolate-spheroidal wave functions provide orthonormal basis which is optimal for the representation of functions whose Fourier transforms are compactly supported [16]. In addition, and to our interest in this manuscript, they require half of the support needed for Gaussian windows to achieve the same target accuracy. Potts et al. [18] used this window function in order to develop a fast summation algorithm and recently Nestler [15] showed that using the Kaiser-Bessel window function, the resulting algorithm is

more accurate than the method using B-splines for homogeneous systems. More recently, Gao et al. [7] used the same window function in their simulation and arrived at the same conclusion for non homogeneous systems. The main drawback of this window function is that it is expensive to compute.

Barnett-Magland window. Recently, Barnett *et al.* [2] introduced an approximation to the Kaiser-Bessel function that does not require expensive Bessel function evaluations. This window function is designated here as the Barnett-Magland (BM) window function and is defined as

$$B(x) = \begin{cases} \frac{e^{\beta\sqrt{1-(\frac{x}{w})^2}}}{e^{\beta}}, & -w \le x \le w, \\ 0, & \text{otherwise.} \end{cases}$$
 (4.5)

As with the Kaiser-Bessel function, this window can be used to achieve high precision. But, unlike the other window functions introduced in this section, its Fourier transform is not known analytically and has to be computed numerically. In section 6.1 we present an approach for efficiently computing the Fourier transform. The BM window function for different β and with w=5 is shown in figure 2. We shall see later that with P points in the support of the window function, i.e. w=P/2, $\beta\approx 2\sqrt{2\pi}w=\sqrt{2\pi}P$ is an optimal value for minimizing approximation errors introduced due to the truncation of window functions and employment of a quadrature rule in the SE method, see section 5.2.

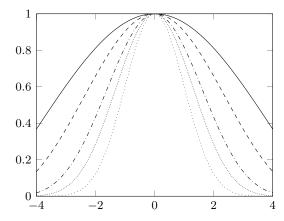
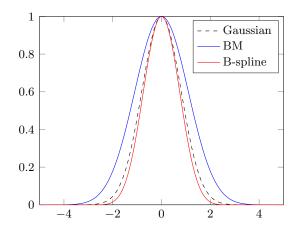


Figure 2: The BM window function with w=5 and $\beta=2,5,10,15,2\sqrt{2\pi}w\approx25$ from top to bottom.

In figure 3 we give an example of the BM window (4.5), Gaussian window (4.1) and B-spline window (4.2) of order p=6, all scaled to one, together with their Fourier spectrum. For the BM window we choose w=6 and $\beta=2\sqrt{2\pi}w$. For the Gaussian window we set $2\xi^2/\eta=1.5\pi/w$ and truncate the window at w=6. The parameters for the Gaussian and BM window functions are selected such that both are truncated at the same level approximately. The left plot of the figure shows that the Gaussian decays faster than the BM window function in real space, whereas the Fourier spectrum of the latter decays much faster in Fourier space.



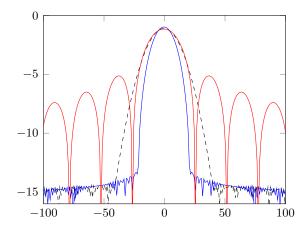


Figure 3: (Left) An example of the BM window (4.5), Gaussian window (4.1) and B-spline (4.2) of order p=6, all scaled to one. (Right) Decay of Fourier spectrum (\log_{10}) of window functions. For BM window, w=6 and $\beta=2\sqrt{2\pi}w$. For Gaussian window, $2\xi^2/\eta=1.5\pi/w$ and the window is truncated at w=6.

5. Errors in the Spectral Ewald method

5.1. Truncation error

Truncation errors in Ewald methods are introduced due to the truncation of interactions in the real space sum or truncation of Fourier modes in the k-space sum. Denoting by ε , as a measure of accuracy, we define the root mean square error (rms) as

$$\varepsilon = \left(\frac{1}{N} \sum_{\mathbf{n}=1}^{N} |\varphi_n - \varphi^*(\mathbf{x_n})|^2\right)^{1/2},$$

where φ^* denotes the accurate or well converged solution of (3.6). The magnitude of the truncation errors can be perfectly estimated using error estimates suggested in [11]. The error estimates suggest that the real space sum decays exponentially as $e^{-r_c^2\xi^2}$ and the k-space sum as $e^{-(\pi k/\xi L)^2}$. Therefore, setting an error tolerance of ε and an Ewald parameter ξ , the cut-off radius is chosen as $r_c \approx \frac{1}{\xi} \sqrt{-\log(\varepsilon)}$ and the maximum wave number as $k_\infty \approx \frac{\xi L}{\pi} \sqrt{-\log(\varepsilon)}$. Assume that ε and ξ are fixed and that the density of the system is constant while N

Assume that ε and ξ are fixed and that the density of the system is constant while N increases, i.e. $N/L^3 = const$. Then, truncating the interactions beyond a certain cut-off, reduces the computational complexity of computing the real space sum from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$ with a constant that depends on the cut-off radius or the number of particles within the cut-off. Applying FFTs, the k-space sum scales as $M^3 \log(M^3)$. On the other hand, we have $k^3 \propto L^3 \propto N$ and therefore, the k-space sum and also the algorithm scale as $N \log(N)$.

5.2. Approximation error

Approximation errors arise due to (a) the evaluation of (3.12) with the trapezoidal rule using truncated window functions and (b) approximating Fourier integrals. We have noted earlier that the quadrature error of the Fourier integrals can be controlled by upsampling the grid using upsampling factors s and s_0 and the parameter n. Therefore, in the rest of this section, we focus only on approximation errors due to (a).

In [12], where Gaussians are used as window functions, authors show that approximation errors can be bounded by

$$C\left(e^{-\pi^2 P^2/(2m^2)} + \operatorname{erfc}(m/\sqrt{2})\right),$$
 (5.1)

in which m is the shape parameter and P is the number of points in the support of each Gaussian in one direction, cf. Theorem 3.1. In (5.1), the first term estimates the quadrature error and the second term estimates the window function truncation error. Balancing both terms, one obtains $m = c\sqrt{\pi P}$, $c \approx 1$. Now, using m, we can find the optimal value of the free parameter, $\eta = P^2 \xi^2 h^2/m^2$.

As with Gaussians, using the BM window function, the accuracy of the algorithm (and not the runtime for a fixed P) strongly depends on the shape parameter β . As suggested in [16], the shape parameter takes the form of $\beta = \pi(1-1/2)P \approx 1.57P$. Here, by numerical experiments, we provide an almost optimal shape parameter which gives the minimum rms error for all P. We heuristically find the following error estimate for approximation errors

$$C\left(\beta^2 e^{-2\pi P^2/\beta} + \operatorname{erfc}(\sqrt{\beta})\right).$$
 (5.2)

Note that this estimate is very similar to the one provided in (5.1). From $\operatorname{erfc}(\sqrt{x}) \approx Ce^{-x}$ for x>0, we observe that, as for the Gaussian window function, the shape parameter is only a function of P, i.e. $\beta=\sqrt{2\pi}P\approx 2.5P$. This implies that approximation errors can be estimated by $Ae^{-2.5P}$ and thereby it can be controlled solely by P. Moreover, in [19], we showed that $A=\sqrt{Q\xi L}/L$, where $Q=\sum_{n=1}^N q_n^2$. Therefore, the error estimates in (5.1) and (5.2) are simplified to $Ae^{-c^2\pi P/2}$ and $Ae^{-\sqrt{2\pi}P}$ respectively.

In figure 4 (left), we plot the rms error in evaluating the triply periodic electrostatic potential (3.6) as a function of β for P=7,10,13, together with the error estimate in (5.2). This figure demonstrates that the optimal β coincides with the minimum of the error curves for each P. Figure 4 (right) shows the rms error as a function of P for different shape parameters as well as the optimal parameter obtained from the error estimate (5.2). The result in this figure suggests that with P=n, almost n digits of accuracy can be achieved. Note that in comparison, Gaussians require $P\approx 2n$ to achieve the same number of digits. Moreover, the optimal selection of β produces exponential convergence of the type e^{-aP} .

6. Parameter selection

There are different parameters present in the SE method and we added some more needed for upsampling and adjusting of window functions and for problems with different periodicities. Here, we provide a systematic approach for selecting these parameters.

Let ξ and ε be given. Using Kolafa and Perram truncation error estimates [11], select r_c and k_{∞} . The grid size in the periodic directions, hence, is chosen as $M=2k_{\infty}$. The error estimates $Ae^{-c^2\pi P/2}$ and $Ae^{-\sqrt{2\pi}P}$ can be used to obtain P, the number of points in the support of the window function (in one direction) for the Gaussian and BM window functions respectively. Hence, window functions are truncated at w=Ph/2 where h=L/M. Based on discussions in the previous section, the shape parameter of window functions are also selected as $m=\sqrt{\pi P}$ (cf. (4.1)) and $\beta=2.5P$ (cf. (4.5)). The upsampling factor in the free-space case and for the zero modes in the 2d- and 1d-periodic cases are selected as $s_0=1+\sqrt{d}$ where d is the number of non periodic directions and the grid size is chosen as $s_0\tilde{M}=s_0(M+P)$. In practice, s_0

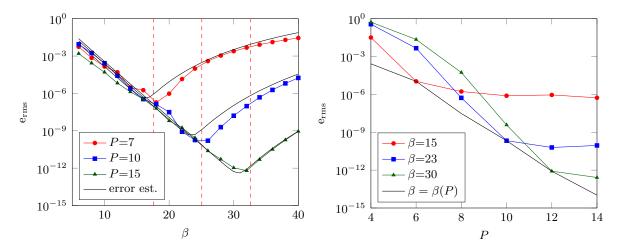


Figure 4: (Left) The rms error and error estimates in evaluating (2.1) with 3d-periodic boundary conditions as a function of β . Also $N=100, L=1, M=28, \xi=6.3$ and C=10. Red dashed lines show $\beta=2.5P$. (Right) The rms error in evaluating (2.1) as a function of the BM window support for different shape parameters. An almost optimal shape parameter depends on P as $\beta=\beta(P)=2.5P$. Also, N=100, L=1, M=28 and $\xi=6.3$.

is adjusted such that the grid size is an even integer. For the 1d- and 2d-periodic cases, we need another upsampling factor for non zero modes. Based on the estimates derived in [19], we choose s such that $e^{-2\pi s} < \varepsilon$. Hence, the grid size is selected as $s\tilde{M} = s(M+P)$. As for s_0 , s has to be adjusted such that $s\tilde{M}$ is an even integer. Finally, the number of Fourier modes to be upsampled is set to $n \approx \lceil 0.1M \rceil$, [19].

6.1. Precomputation

For BM window functions, Fourier transforms can be precomputed and reused. In the precomputation step, we require at most three 1D FFTs and the tensorial product to compute the Fourier transform. Using the known fast Gaussian gridding approach, we can also reduce the cost of evaluating Gaussians, see [12] and references therein.

We have already discussed that the calculation of zero modes can be accelerated by the introduction of AFT/AIFT for the 2d- and 1d-periodic cases. However, this idea cannot be applied for the free-space problem. In this case, the upsampling factor is $s_0 \geq 1 + \sqrt{3}$ and as an example for M = 100, and P = 16, the grid has to be extended $(s_0(M+P))^3/M^3 \approx 32$ times. Let us remind that we require this upsampling factor in order to resolve the oscillatory mollified Green's functions (3.2)-(3.5) and to compute an aperiodic convolution. By precomputing an effective Green's function and truncating it, we only require zero padding by a factor of 2 in each direction to carry out the convolution. We require the following steps, [1],

- 1. Calculate \widehat{G} on a $(s_0 \widetilde{M})^3$ grid.
- 2. Apply a 3d IFFT to compute G.
- 3. Truncate G to obtain G_R .
- 4. Apply a 3D FFT to get \widehat{G}_R .

Then, $\widehat{G}_{\scriptscriptstyle R}$ can be stored and reused in the scaling step.

7. Numerical results

Here, we provide numerical results for computing the Fourier space part of the electrostatic potential in (3.6) using the Spectral Ewald method with Gaussian and BM window functions.

In this section, we compute the relative rms error. All experiments in this work are performed with a single core on a machine with Intel Core i7-3770 CPU which runs on 3.40 GHz with 8 GB of memory. The unified Spectral Ewald code is publicly available online [14]. The core routines are written in C and are dynamically linked and called through a MATLAB MEX interface. Moreover, the package is built with the GNU C Compiler at version 5.4.0.

In the first numerical experiment, we compare Gaussian and BM window functions in terms of cost and accuracy. We choose a system of $N = 10^5$ uniformly distributed particles in a unit box with $\xi = 6.3$ and M = 28. The rest of the parameters are chosen to be the same for both window functions and set such that all errors due to upsampling needs are negligible.

In figure 5 (left), we plot the relative rms error as a function of P, where P^3 is the number of points in the support of the window function. The result shows that machine precision can be achieved using BM window functions with P=16 whereas we require P=24 using the truncated Gaussian. This difference however is smaller for low accuracy demands.

The runtime as a function of P is demonstrated in figure 5 (right) for both window functions. We use the same system as in the left figure and only include the runtime of the spreading and gathering steps. The plot manifests that for a fixed P, the runtime is approximately the same regardless of which of the two window functions that is utilized.

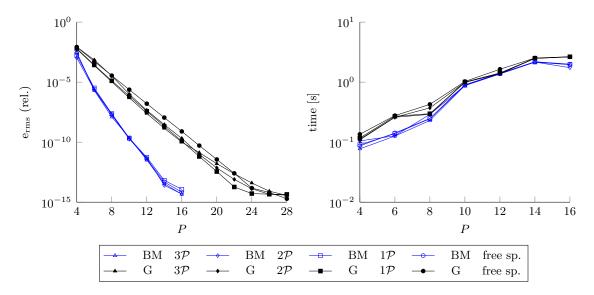


Figure 5: (Left) Relative rms error and (right) runtime as a function of P using BM and Gaussian windows for the 3d-, 2d-, 1d-periodic and free-space cases. $N=10^5, L=1, \beta=2.5P$. Also $\xi=6.3$ and M=28. The other parameters are chosen such that other errors are negligible. The runtime includes only the cost of gridding and gathering steps, see Algorithm 3.

To study the effect of periodicity on the computational cost of the SE method, let us consider a uniformly distributed system of $N=10^5$ particles in a box of size L=10. The Ewald parameter is selected such that the runtime of the real space and Fourier space sums for the triply periodic case are balanced. As a result we obtain $\xi=3$. Note that this selection heavily depends on the implementation of both sums and the hardware that is used for the simulation. For each error tolerance, the grid size M (in periodic directions) is chosen based on the Kolafa & Perram truncation error estimate [11] and regardless of the periodicity and window functions. The other parameters are selected as discussed in section 6.1. In figure 6 we illustrate the Fourier space runtime (excluding the precomputation) of the AFT+scaling (left)

and spreading+gathering (right) steps separately. The runtimes in the left plot are mainly functions of the grid size and periodicity while the results in the right plot depend only on N and the number of points in the support of window functions. It is evident that, for cases with the same type of boundary conditions, the runtime curves of the AFT+scaling steps are the same. Moreover, as the number of non periodic directions increases, the total number of grid points and consequently the cost of the AFT+scaling steps grows. Nonetheless, the AFT algorithm reduces the runtime significantly, compared to the full upsampling, such that the 2d-and 3d- periodic cases have similar computational cost. Also, comparing the runtimes of the free-space and triply periodic cases, we see the effect of the extra upsampling factor of two in all three non-periodic directions.

The right plot in figure 6 demonstrates the runtime of the spreading+gathering steps as a function of the relative rms error using both window functions and for different types of periodicities. We remind that in the gridding and gathering steps, window functions are used to interpolate charges to and from the uniform grid. The figure clearly shows that this cost reduced when the BM window function is used. Studying the data, we note that for each error tolerance in this example, we require approximately $P_{Gaussian} = 2P_{BM}$.

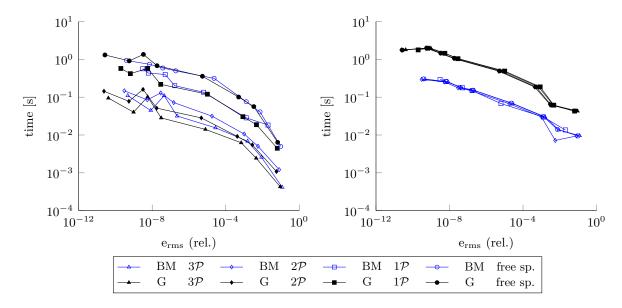


Figure 6: Runtime of (left) the FFT and scaling steps and (right) the spreading and gathering steps vs relative rms error. The SE method with the BM and Gaussian window functions with D=0,1,2,3 periodic dimensions is used to compute the Fourier space part of the potential. $N=10^5, L=10, \beta=2.5P$ and $\xi=3$. Also $M=32,\ldots,90$ and solely obtained via truncation error estimates. Precomputation is excluded from runtimes.

Next, we add up the runtimes in both plots to obtain the total runtimes, as shown in figure 7. In [19] we showed that, due to the employment of AFTs, the cost of 1d-periodic and 3d-periodic cases are only marginally different. Clearly, this is also true for the 2d-periodic case which is in general less expensive than the 1d-periodic case but more expensive than the 3d-periodic case. Here, we observe that the effect of periodicity is further diminished such that the doubly periodic and triply periodic cases have a very similar cost for both window function. The singly periodic case is only up to two times (for very strict error tolerances) more expensive compared to the fully periodic case when using the BM window function. Since the spreading/gathering step is more costly using Gaussians, the relative total runtime is in this

case much less affected by periodicity. For the free-space case a precomputation step reduces the cost of FFT computations. In figure 7 we observe that, as a result, the free-space case is up to four times more expensive than the triply periodic case to achieve high accuracy.

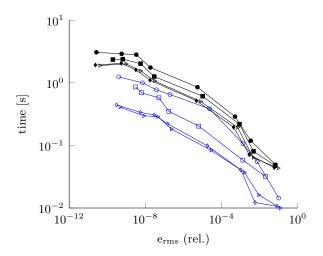


Figure 7: Runtime vs relative rms error in computing the Fourier space part of the potential using the BM and Gaussian window functions with arbitrary periodicity. $N=10^5, L=10, \beta=2.5P$ and $\xi=3$. Also $M=32,\ldots,90$ and solely obtained via truncation error estimates. Precomputation is excluded from the runtime. Legends are the same as in figure 6.

8. Conclusions

In this work, we presented a unified approach in the calculation of the Fourier space part of the Ewald sum in the task of computing the electrostatic potential with free-space, singly, doubly and triply periodic boundary conditions. We used the Spectral Ewald method together with the very recent idea of Vico et al. [21] in order to unify the treatment of all Fourier modes and to utilize FFTs to accelerate the calculation. This approach has been already used for the 1d-periodic [19] and free-space cases [1] but not for the 2d-periodic case. In an attempt to compute zero Fourier modes in this case, Lindbo and Tornberg [13] used Chebyshev interpolations and showed that the extra cost of calculating these modes is small compared to the total cost of the SE method. In this paper we unified these modes into the treatment of the other modes and completed the framework.

We also extended the idea of the adaptive Fourier transform, first developed in [19], for the 2d-periodic case. With this, we were able to apply upsampling on only 20 percent of Fourier modes and reduce further the cost of calculations.

We compared Gaussians, previously used in the SE method, with a new window function, introduced recently by Barnett et al. [2]. This window, denoted here as BM window function, is a simplification to the original Kaiser-Bessel window function. The BM window accelerates the spreading and gathering steps of the method by reducing the number of points in the support as compared to the Gaussian window function to achieve the same error level. Using the BM window function, machine accuracy can be achieved with half of the support needed for the Gaussian window function. As for Gaussians, the new window function maintains spectral accuracy and the method scales as $\mathcal{O}(N \log(N))$. Implementing the new window function in

the SE method, we compared the resulting method with the existing algorithm that has been developed in a series of papers [12, 13, 19, 1].

To compute the optimal shape parameter of the new window function, we numerically estimated the approximation error due to the truncation of the modified Kaiser-Bessel window function and the employment of the trapezoidal rule. We then showed that the optimal shape parameter and therefore the approximation error can be controlled using a single parameter, namely the number of points in the support of the window function.

Our numerical experiments showed that that the AFT algorithm can reduce the runtime remarkably such that the cost of computing doubly periodic and triply periodic cases are very similar and the singly periodic case is only up to two times more expensive, the most for very strict error tolerances. In order to accurately compute Fourier integrals in the free-space case, the grid has to be upsampled up to two times in each direction. However, our results indicated that, by precomputing Fourier coefficients of the mollified Green's function, the free-space case is at most four times more expensive than the triply periodic case.

The unified Spectral Ewald package to solve electrostatic problems with different boundary conditions is accelerated using OpenMP and vector intrinsics and will be soon available online at [14].

9. Acknowledgment

This work has been supported by the Göran Gustafsson Foundation for Research in Natural Sciences and Medicine and by the Swedish e-Science Research Center (SeRC). The authors gratefully acknowledge this support.

References

- [1] L. af Klinteberg, D. S. Shamshirgar, and A.-K. Tornberg. Fast Ewald summation for free-space Stokes potentials. *Res. Math. Sci.*, 4(1):1, 2017.
- [2] A. H. Barnett and J. F. Magland. FINUFFT: a fast and lightweight non-uniform fast Fourier transform library. *Prep.*, 2017, URL https://github.com/ahbarnett/finufft.
- [3] M. Deserno and C. Holm. How to mesh up Ewald sums. I. A theoretical and numerical comparison of various particle mesh routines. *J. Chem. Phys.*, 109(18):7678–7693, 1998.
- [4] U. Essmann, L. Perera, M. L. Berkowitz, T. Darden, H. Lee, and L. G. Pedersen. A smooth particle mesh Ewald method. *J. Chem. Phys.*, 103(19):8577–8593, 1995.
- [5] P. P. Ewald. Die Berechnung optischer und elektrostatische Gitterpotentiale. Ann. Phys., 64:253–287, 1921.
- [6] J. G. Fripiat, J. Delhalle, I. Flamant, and F. E. Harris. Ewald-type formulas for Gaussian-basis Bloch states in one-dimensionally periodic systems. J. Chem. Phys., 132(4):044108, 2010.
- [7] X. Gao, J. Fang, and H. Wang. Kaiser-Bessel basis for particle-mesh interpolation. *Phys. Rev. E*, 95(6):063303, 2017.

- [8] A. Grzybowski, E. Gwóźdź, and A. Bródka. Ewald summation of electrostatic interactions in molecular dynamics of a three-dimensional system with periodicity in two directions. *Phys. Rev. B*, 61(10):6706–6712, 2000.
- [9] F. Harris. On the use of windows for harmonic analysis with the discrete Fourier transform. *Proc. IEEE*, 66(1):51–83, 1978.
- [10] R. Hockney and J. Eastwood. Computer Simulation Using Particles. McGraw-Hill, New York, 2010. ISBN 1439822050.
- [11] J. Kolafa and J. W. Perram. Cutoff Errors in the Ewald Summation Formulae for Point Charge Systems. *Mol. Simul.*, 9(5):351–368, 1992.
- [12] D. Lindbo and A.-K. Tornberg. Spectral accuracy in fast Ewald-based methods for particle simulations. *J. Comput. Phys.*, 230(24):8744–8761, 2011.
- [13] D. Lindbo and A.-K. Tornberg. Fast and spectrally accurate Ewald summation for 2-periodic electrostatic systems. *J. Chem. Phys.*, 136(16), 2011.
- [14] D. Lindbo, L. af Klinteberg, and D. Saffar Shamshirgar. The Spectral Ewald Unified package, 2016. URL http://github.com/davoudss/SE_unified.
- [15] F. Nestler. Parameter Tuning for the NFFT Based Fast Ewald Summation. Front. Phys., 4, 2016.
- [16] A. Osipov, V. Rokhlin, and H. Xiao. Prolate Spheroidal Wave Functions of Order Zero: Mathematical Tools for Bandlimited Approximation. Springer, 2013.
- [17] M. Porto. Ewald summation of electrostatic interactions of systems with finite extent in two of three dimensions. J. Phys. A. Math. Gen., 33(35):6211–6218, 2000.
- [18] D. Potts and G. Steidl. Fast Summation at Nonequispaced Knots by NFFT. SIAM J. Sci. Comput., 24(6):2013–2037, 2003.
- [19] D. Saffar Shamshirgar and A.-K. Tornberg. The Spectral Ewald method for singly periodic domains. *J. Comput. Phys.*, 347:341–366, 2017.
- [20] A.-K. Tornberg. The Ewald sums for singly, doubly and triply periodic electrostatic systems. *Adv. Comput. Math.*, 42(1):227–248, 2016.
- [21] F. Vico, L. Greengard, and M. Ferrando. Fast convolution with free-space Green's functions. *J. Comput. Phys.*, 323:191–203, 2016.