A particle-particle, particle-density algorithm for the calculation of electrostatic interactions of particles with slablike geometry

S. Alireza Ghasemi, Alexey Neelov, and Stefan Goedecker

Citation: The Journal of Chemical Physics **127**, 224102 (2007); doi: 10.1063/1.2804382
View online: http://dx.doi.org/10.1063/1.2804382
View Table of Contents: http://scitation.aip.org/content/aip/journal/jcp/127/22?ver=pdfcov
Published by the AIP Publishing

A particle-particle, particle-density algorithm for the calculation of electrostatic interactions of particles with slablike geometry

S. Alireza Ghasemi,[a] Alexey Neelov, and Stefan Goedecker[b]
Condensed Matter Theory Group, Department of Physics, University of Basel, Klingelbergstrasse 82, Basel 4056, Switzerland

(Received 5 July 2007; accepted 10 October 2007; published online 11 December 2007)

We present a fast and accurate method to calculate the electrostatic energy and forces of interacting particles with the boundary conditions appropriate to surfaces, i.e., periodic in the two directions parallel to the surface and free in the perpendicular direction. In the spirit of the Ewald method, the problem is divided into a short range and a long range part. The charge density responsible for the long range part is represented by plane waves in the periodic directions and by finite elements in the nonperiodic direction. Our method has computational complexity of $\mathcal{O}(N_g \log(N_g))$ with a very small prefactor, where N_g is the number of grid points. © *2007 American Institute of Physics*. [DOI: 10.1063/1.2804382]

I. INTRODUCTION

Simulations of systems with slablike geometries are of great importance. Problems involving surfaces, interfaces, tip-surface interaction in scanning probe microscopy simulations, electrolytes trapped between two plates, thin films of ferrofluids, etc., all fall into this category. Calculating the Coulomb interactions in such setting is a major challenge. With free boundary condition (i.e., the potential tends to be zero at infinity), the scaling of the trivial direct summation is $\mathcal{O}(N^2)$, where N is the number of particles. In the case of two-dimensional (2D) periodic and one-dimensional (1D) free (2DP1DF) boundary conditions (BCs), the situation is even worse. In principle, one would then have to include into the summation the interactions with all the periodic images in the two periodic directions.

Algorithms such as Ewald-based methods,[1] fast multipole methods (FMMs),[2] particle-particle, particle-mesh (P^3M) method,[3] and convergence factor approaches[4–6] have, therefore, been generalized to 2DP1DF problems. Handling different types of BCs in FMM (Ref. 7), is straightforward. In addition, the FMMs have the ideal linear scaling. Unfortunately, the prefactors in FMMs are typically large and so, the FMM methods are, in many cases, only faster than other methods for $N > 10^6$, where N is the number of particles. Another drawback of FMM that is important in molecular dynamics is that the approximate FMM forces are not analytical derivatives of the approximate energy. Therefore, the energy is not conserved during the molecular dynamics simulation. High accuracy energy conservation is, therefore, impossible.

Ewald methods for 2DP1DF boundary conditions, called EW2D, have been developed in Refs. 8–10. A comparison of three versions of EW2D method can be found in Ref. 11. Unfortunately, the practical use of the EW2D sum is hampered by the occurrence of a reciprocal space term. The resulting Fourier space sum does not allow for a product decomposition as it is done in the three-dimensional (3D) periodic Ewald method and, therefore, the method has a scaling of $\mathcal{O}(N^2)$. In 2002, Arnold and Holm developed MMM with 2DP1DF BC (MMM2D),[12] which is found to be the best in terms of accuracy. Another advantage of this method is that it has "*a priori*" error estimates. However, because of its $\mathcal{O}(N^{5/3})$ scaling, it is only suitable for a small number of atoms.

A rather simple approach is to use the standard three-dimensional periodic Ewald method (EW3D) also for 2DP1DF boundary conditions. Spohr showed that the regular EW3D method almost reproduces the EW2D results,[13] provided that the box length in the nonperiodic direction is about five times larger than those in the periodic directions and that there is an empty space of sufficient thickness in the basic periodic box to dampen out the interslab interactions. There are also methods with correction terms to make the 3D periodical scheme applicable to the 2DP1DF systems and to resolve the problem of slow convergence with respect to thickness, so that a medium size gap (empty space) is enough. The EW3DC (Refs. 14 and 15) method consists of a modification of EW3D to account for the slab geometry and an addition of a correction term to remove the forces due to the net dipole of the periodically repeating slabs. Methods with layer correction terms to eliminate the interslab interaction, in addition to the correction term responsible for the net dipole, have been mixed with mesh-based methods; thus, almost linear scaling is achieved, e.g., three dimensional Ewald-type method with layer correction term [EW3DLC (Refs. 16 and 17)] and particle-particle, particle-mesh method with layer correction term [P^3MLC (Refs. 16 and 17)]. The main drawback of these methods is that the errors in the forces on the particles near the surfaces are more than in the middle. Although the error becomes negligible by choosing proper values of the empty space,[18] this reduces the efficiency of the method. An interesting feature of EW3DLC

[a]Electronic mail: alireza.ghasemi@unibas.ch
[b]URL: http://pages.unibas.ch/comphys/comphys/

is that there is *a priori* error estimate for the layer correction term.

In this paper, we present a method which fills the gap of absence of an efficient method for medium size systems having 10^2-10^6 particles. Because our method is not based on a modification of a fully periodic method, no replication is needed in the nonperiodic direction, leading to smaller memory and CPU requirements. In contrast to some other methods, our method does not impose any restriction on the distribution of particles in the nonperiodic direction.

II. COULOMB INTERACTION FOR SYSTEMS WITH 2DP1DF BC

Consider a system of N particles with charges q_i at positions $\mathbf{r}_i$ in an overall neutral and rectangular simulation box of dimensions L_x, L_y, and L_z. The Coulomb potential energy of this system with periodic boundary condition in two directions and free boundary conditions in the third direction (let us say in the z direction) can be written as

$$E = \frac{1}{2}\sum_{\mathbf{n}}{}' \sum_{i,j=1}^{N} \frac{q_i q_j}{|\mathbf{r}_{ij}+\mathbf{n}|}, \tag{1}$$

where $\mathbf{r}_{ij}=\mathbf{r}_i-\mathbf{r}_j$ and $\mathbf{n}=(n_x L_x, n_y L_y, 0)$, with n_x and n_y being integers. The prime on the outer sum denotes that for $\mathbf{n}=0$, the term $i=j$ has to be omitted.

In the Ewald-type methods, the above very slowly converging sum over the Coulomb potential function is split into two sums which converge exponentially fast, one in real space and the other in the Fourier space. This splitting can be done by adding and subtracting a term corresponding to the electrostatic energy of a system of smooth spherical charge densities $\rho_i(\mathbf{r})$ centered on the particle positions,

$$\begin{aligned} E = {} & \frac{1}{2}\sum_{\mathbf{n}}{}' \sum_{i,j=1}^{N} \left[\frac{q_i q_j}{|\mathbf{r}_{ij}+\mathbf{n}|} - \int\int \frac{\rho_i(\mathbf{r})\rho_j(\mathbf{r}'+\mathbf{n})}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \right] \\ & + \frac{1}{2}\sum_{\mathbf{n}} \sum_{i,j=1}^{N} \int\int \frac{\rho_i(\mathbf{r})\rho_j(\mathbf{r}'+\mathbf{n})}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r} d\mathbf{r}' \\ & - \frac{1}{2}\sum_{i=1}^{N} \int\int \frac{\rho_i(\mathbf{r})\rho_i(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r} d\mathbf{r}'. \end{aligned} \tag{2}$$

The aim of the last term is to subtract the self-energy for $\mathbf{n}=0$ and $i=j$, which is included in the second term.

Even though Ewald-type methods allow for any choice of $\rho_i(r)$, it was noted in Refs. 19 and 20 that Gaussians are virtually optimal in practice. By choosing $\rho_i(r)$ to be a Gaussian function,

$$\rho_i(\mathbf{r}) = \frac{q_i}{(\alpha^2\pi)^{3/2}} \exp\left[-\frac{|\mathbf{r}-\mathbf{r}_i|^2}{\alpha^2} \right], \tag{3}$$

the first and third terms in Eq. (2) can be rewritten as follows:[20]

$$\begin{aligned} E = {} & \frac{1}{2}\sum_{\mathbf{n}}{}' \sum_{i,j=1}^{N} \frac{q_i q_j \operatorname{erfc}\left[\frac{|\mathbf{r}_{ij}+\mathbf{n}|}{\alpha\sqrt{2}} \right]}{|\mathbf{r}_{ij}+\mathbf{n}|} \\ & + \frac{1}{2}\sum_{\mathbf{n}} \sum_{i,j=1}^{N} \int\int \frac{\rho_i(\mathbf{r})\rho_j(\mathbf{r}'+\mathbf{n})}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r} d\mathbf{r}' - \frac{1}{\alpha\sqrt{2\pi}} \sum_{i=1}^{N} q_i^2. \end{aligned} \tag{4}$$

Obviously, the calculation of the third term is trivial. Since the interaction in the first term is decaying exponentially, it can be made of finite range by introducing a cutoff. The error resulting from the cutoff is then also exponentially small and the short range term can be calculated with linear scaling. We have calculated the short range part and also the contribution of forces from long range as it is described in Ref. 19.

The major difficulty is the calculation of the second term. A method to solve Poisson's equation under 2DP1DF boundary conditions has recently been put forward by Genovese *et al.*[21] Our approach is similar. As in Ref. 21, we use plane waves[22] to represent the charge density in the periodic directions. Whereas Genovese *et al.* used scaling functions as the basis in the nonperiodic direction, we use finite elements for that purpose. Scaling functions are presumably the optimal choice in the context of electronic structure calculations, where the charge density is given on a numerical grid. In our case, the charge distribution is a sum over smooth Gaussians that can easily be represented by our mixed basis set of plane waves and finite elements. As will be seen, we can avoid storing any kernel if we solve a differential equation along the z axis instead of solving an integral equation. We will use a family of finite elements that allows us to solve the linear system of equations resulting from the differential equation very efficiently.

A. Calculating the long range part

The second term in Eq. (4) can be written as

$$E_{\text{long}} = \frac{1}{2}\int_{\Re^3} \rho^{(N)}(\mathbf{r}) V(\mathbf{r}) d\mathbf{r}, \tag{5}$$

where

$$\rho^{(N)}(\mathbf{r}) := \sum_{i=1}^{N} \rho_i(\mathbf{r}), \tag{6a}$$

$$V(\mathbf{r}) := \int_{\Re^3} \frac{\rho(\mathbf{r}')}{|\mathbf{r}-\mathbf{r}'|} d\mathbf{r}', \tag{6b}$$

$$\rho(\mathbf{r}) := \sum_{\mathbf{n}} \sum_{j=1}^{N} \rho_j(\mathbf{r}+\mathbf{n}). \tag{6c}$$

We consider a system with a charge density that is only localized in the nonperiodic direction, in our notation z; $\rho(x,y,z)=0\,\forall\,(x,y,z)\in\Re^3 | z\notin[z_l,z_u]$. We define the cell containing the continuous charge density as

$$\mathcal{V} := [0,L_x] \otimes [0,L_y] \otimes [z_l,z_u].$$

In our case, the length of $\mathcal{V}$ in z direction $z_u - z_l$ is L_z plus twice the cutoff for Gaussians. Since $\rho(\mathbf{r})$ is periodic in x and y directions, $V(\mathbf{r})$ is periodic too, so we can rewrite Eq. (5) as

$$E_{\text{long}} = \frac{1}{2}\int_{\mathcal{V}} \rho(\mathbf{r})V(\mathbf{r})d\mathbf{r}, \tag{7}$$

and $V(\mathbf{r})$ can be calculated in an alternative way to Eq. (6b). It can be considered as the solution of Poisson's equation with 2DP1DF BC,

$$\nabla^2 V(\mathbf{r}) = -4\pi\rho(\mathbf{r}). \tag{8}$$

In order to calculate the contribution of the forces resulting from the long range part, the following equation is used:

$$\mathbf{F}_i^{(\text{long})} = \frac{1}{2}\int_{\mathcal{V}} \zeta_i(\mathbf{r})V(\mathbf{r})d\mathbf{r} \approx \frac{1}{2}\sum_{klm} \zeta_{klm}V_{klm}, \tag{9}$$

where $\zeta_i(\mathbf{r})$ is the derivative of Gaussian charge distribution with respect to particle position $\mathbf{r}_i$; more details can found in Ref. 19. The charge density and the potential are periodic in x and y directions. Hence, we can write the potential and the charge density in terms of Fourier series,

$$V(x,y,z) = \sum_{k,l=-\infty}^{\infty} c_{kl}(z)\exp\left[2i\pi\left(\frac{kx}{L_x}+\frac{ly}{L_y}\right)\right], \tag{10a}$$

$$\rho(x,y,z) = \sum_{k,l=-\infty}^{\infty} \frac{\eta_{kl}(z)}{-4\pi}\exp\left[2i\pi\left(\frac{kx}{L_x}+\frac{ly}{L_y}\right)\right]. \tag{10b}$$

Inserting Eqs. (10a) and (10b) in Eq. (8) yields

$$\left(\frac{d^2}{dz^2} - g_{kl}^2\right)c_{kl}(z) = \eta_{kl}(z), \tag{11}$$

$$\begin{aligned} g_{kl} &:= 2\pi\sqrt{\frac{k^2}{L_x^2}+\frac{l^2}{L_y^2}}, \\ \eta_{kl}(z) &= \frac{-4\pi}{L_xL_y}\int_0^{L_x}\int_0^{L_y}\rho(x,y,z) \\ &\quad\times\exp\left[-2i\pi\left(\frac{kx}{L_x}+\frac{ly}{L_y}\right)\right]dxdy. \end{aligned} \tag{12}$$

To solve the differential equation (11), one needs to have boundary conditions at $z\to\pm\infty$ for $c_{kl}(z)$. The potential obtained by solving Poisson's equation should be the same as the one in Eq. (6b). Hence, we derive the boundary condition in the nonperiodic direction from Eq. (6b). Considering the facts that the charge density is supposed to be limited in the nonperiodic direction and that we search for the boundary condition at $z\to\pm\infty$ yields $|\mathbf{r}-\mathbf{r}'|\neq 0$. By performing the Taylor expansion of $1/|\mathbf{r}-\mathbf{r}'|$ at about $z'=0$ in the integral expression of Eq. (6b) for the exact potential $V(x,y,z)$ arising from our periodic charge distribution $\rho(\mathbf{r})$,

$$\begin{aligned} V(x,y,z) &= \int_{z_l}^{z_u}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} dx'dy'dz'\frac{1}{|\mathbf{r}-\mathbf{r}'|} \\ &\quad\times\sum_{k,l=-\infty}^{\infty}\frac{\eta_{kl}(z')}{-4\pi}\exp\left[2\pi i\left(\frac{kx'}{L_x}+\frac{ly'}{L_y}\right)\right], \end{aligned} \tag{13}$$

one can show that $V(x,y,z\to\pm\infty) = \mp\beta$, where β is proportional to the dipole moment of the charge distribution along the z direction,

$$\beta = \frac{1}{2}\int_{z_l}^{z_u}\eta_{00}(z')z'dz'. \tag{14}$$

For the Gaussian charge distributions given by Eq. (3), the above integral can be calculated analytically and β is calculated exactly,

$$\beta = \frac{-2\pi}{L_xL_y}\sum_{i=1}^{N} q_iz_i. \tag{15}$$

This boundary condition for the potential gives the following conditions for the g's:

- $g=g_{00}=0 \Rightarrow (d^2/dz^2)c_{00}(z)=\eta_{00}(z)$, we solve this differential equation with boundary condition $c_{00}(z\to\pm\infty) = \mp\beta$; and
- $g=g_{kl}\neq 0 \Rightarrow (d^2/dz^2 - g_{kl}^2)c_{kl}(z)=\eta_{kl}(z)$, for all of these differential equations, we have to impose BC of the form $c_{kl}(z\to\pm\infty)=0$.

The solution for $c_{00}(z)$ is a linear function outside the interval $[z_l,z_u]$. Since the boundary conditions are applied at infinity, the linear term has to vanish and one has to satisfy Dirichlet BCs for c_{00}, namely, $c_{00}(z_u)=-\beta$ and $c_{00}(z_l)=\beta$. For $|k|+|l|>0$, $c_{kl}(z)$ will have Robin BC, as explained below. The potential is thus not modified if one takes, for instance, a computational box that is thicker in the z direction than necessary. The thinnest possible box is the one that just includes the region where the charge is nonzero.

For $z\in(-\infty,z_l]$, we have $\eta_{kl}(z)=0$; thus, it yields

$$c(z) = c(z_l)e^{g_{kl}(z-z_l)}. \tag{16}$$

Both $c(z)$ and its derivative must be continuous. So, performing left differentiation at z_l, we get

$$c'(z_l) - g_{kl}c(z_l) = 0. \tag{17}$$

With a similar procedure, we obtain the BC at z_u,

$$c'(z_u) + g_{kl}c(z_u) = 0. \tag{18}$$

These BCs are in agreement with the BCs resulting from Green's functions in Ref. 21.

B. Solving the ordinary differential equation using the finite element method

We recapitulate the procedure of solving the differential equation for the case of $|k|+|l|>0$, i.e., $g_{kl}\neq 0$, using the finite element method. For the case of $k=l=0$, the approach is similar, with the only difference that the Dirichlet BCs are used. The case of $k=l=0$ can be found in many manuscripts and textbooks on the finite element method, e.g., Ref. 23. In

particular, our notation follows Ref. 23. Discretizing the differential equation with mentioned Robin BCs using the finite element method leads to a system of linear equations. The resulting matrix is a banded matrix for which the system of equations can be solved efficiently if high-order hierarchical piecewise polynomials are used as a basis and if the degrees of freedom are decimated. This hierarchical finite element basis set leads to algebraic systems that are less susceptible to round off error accumulation at high order than those produced by a Lagrange basis.[24] We use linear hat functions as the linear hierarchical basis. For higher order bases, we exploit the method of Szabo and Babuska,[25] which relies on Legendre polynomials. Below, we show the expansion of $c(z)$ in terms of the hat functions and the other higher order hierarchical piecewise polynomials on the interval $[z_{i-1}, z_i]$,

$$c(z) \approx c_{i-1}N_{-1}(\xi_i) + c_i N_1(\xi_i) + \sum_{j=2}^{p} c_{i,j}N_j(\xi_i), \tag{19}$$

where $\xi_i = 2(z-z_i)/h+1$, $h=z_i-z_{i-1}$, and the functions $N_i(\xi)$ in the interval $[-1,1]$ are given by

$$N_{-1}(\xi) = \frac{1-\xi}{2}, \quad N_1(\xi) = \frac{1+\xi}{2}, \tag{20a}$$

$$N_i(\xi) = \sqrt{\frac{2i-1}{2}} \int_{-1}^{\xi} P_{i-1}(\xi')d\xi', \quad i \geqslant 2, \tag{20b}$$

where $P_i(\xi)$ are Legendre polynomials. These hierarchical bases have useful orthogonality properties that lead to sparse and well-conditioned stiffness matrices. Defining an operator $\mathcal{L}$,

$$\mathcal{L}[c] \coloneqq c''(z) - g^2 c(z), \tag{21}$$

we can write our differential equation (11) as

$$\mathcal{L}[c] = \eta(z),$$

with boundary conditions

$$\begin{aligned} c'(z_l) - gc(z_l) &= 0, \\ c'(z_u) + gc(z_u) &= 0. \end{aligned} \tag{22}$$

The method of weighted residuals is used to construct a variational integral formulation of Eq. (21) by multiplying with a test function $d(z)$ and integrating over $[z_l, z_u]$,

$$(d, \mathcal{L}[c] - \eta) = 0, \quad \forall\, d \in H^1(z_l, z_u), \tag{23}$$

where H^1 is the Sobolev space. We have introduced the L^2 inner product,

$$(d,c) \coloneqq \int_{z_l}^{z_u} d(z)c(z)dz. \tag{24}$$

Performing the integration by parts in Eq. (23) and applying Robin BCs given in Eq. (22) give

$$A(d,c) = (d,\eta) + gd(z_l)c(z_l) + gd(z_u)c(z_u), \tag{25}$$

where

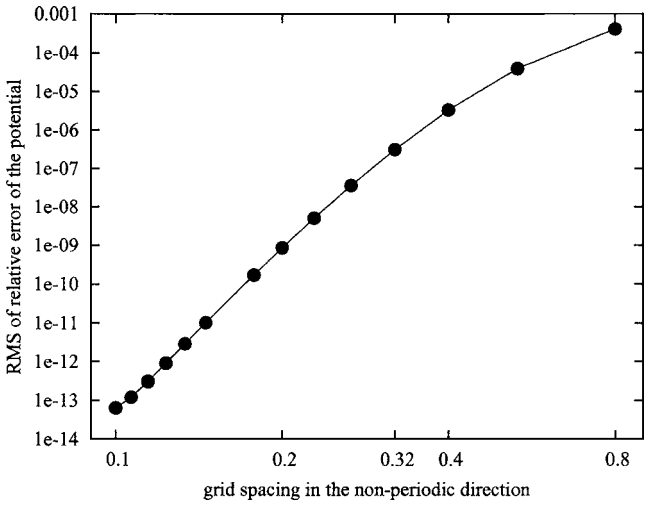

FIG. 1. rms of relative error for the potential given in Sec. III A with $a=10$, $b=10$, $c=1$, and $L_x=L_y=1$. On this double logarithmic plot, the curve has an asymptotic slope of 14 and machine precision can be reached.

$$A(d,c) \coloneqq \int_{z_l}^{z_u} [-d'(z)c'(z) - g^2 d(z)c(z)]dz. \tag{26}$$

Using the Galerkin approach and exploiting the decimation scheme, we can construct a system of linear equations $B\mathbf{c}=\mathbf{b}$, where the elements of the vector $\mathbf{c}$ are the values of $c(z)$ at grid points. The detailed structure of this linear system of equations is given in the Appendix.

In practice, we put the charge density, the collection of the Gaussian charge distributions, on the mesh $n_x \times n_y \times n_z$ and by performing n_z two-dimensional fast Fourier transform, we obtain $\eta_{kl}(z)$ on the grid points. The time of calculation of this part scales $n_x n_y n_z \log(n_x n_y)$; the inverse of this part, i.e., to calculate potential function from $c_{kl}(z)$, scales the same. To calculate $c_{kl}(z)$ from $\eta_{kl}(z)$, we solve $n_x n_y$ system of linear equations that the corresponding matrix is tridiagonal, so that this part can be solved with a small prefactor and with the complexity of $n_x n_y n_z$. Thus, the method scales $N_g \log(N_g)$, where $N_g = n_x \times n_y \times n_z$.

III. NUMERICAL RESULT

In this section, we present the numerical results obtained for Poisson's solver for continuous charge densities with 2DP1DF BC in stand-alone mode and for our Ewald-type method for point particles interacting by Coulombic potential with 2DP1DF BC. We also show numerical evidence for the conservation of energy in molecular dynamics simulation of a system composed of sodium chloride atoms.

A. Numerical results for the Poisson's solver

Our method has an algebraic convergence rate in the nonperiodic direction and a faster exponential convergence rate in the periodic directions, respectively, due to the finite element polynomial bases and to the plane wave representation. In Fig. 1, we show the convergence rate in nonperiodic direction with seventh order finite elements [$p=7$ in Eq. (19)]; this value is used throughout this manuscript. For our test, the starting point was the potential rather than the

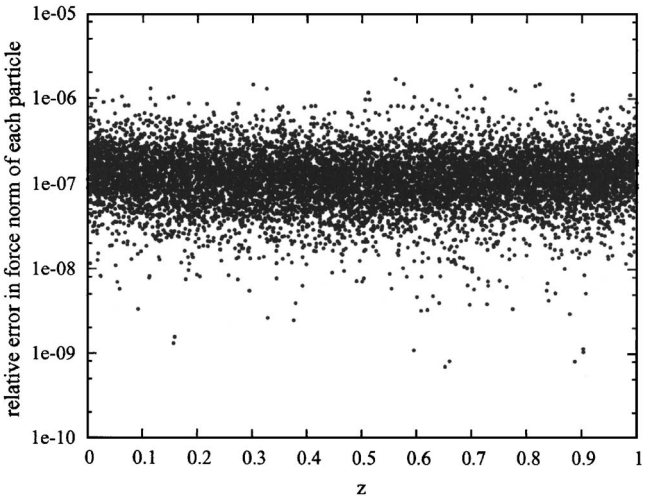

FIG. 2. Relative error distribution of force norm on each particle along the z axis for 100 random systems with 100 particles.

charge density, since the charge density can be obtained analytically from the potential by simple differentiation. Our test potential had the form $V(\mathbf{r})=\sin(a\sin(2\pi x/L_x))\times\sin(b\sin(2\pi y/L_y))\exp(-z^2/c^2)$.

B. Numerical results for point particles

In this section, we give the numerical results of our implementation of the presented method for point particles. Since MMM2D is known to be highly accurate, we use it as a reference in this section. First, we want to demonstrate that error distribution along the nonperiodic direction is uniform unlike in the 3D periodic methods with correction terms.[14,16,17] To this aim, 100 particles were put randomly in a unit cubic cell and the program was run 100 times each time with different random positions. Results of the relative error of forces exerted on each particle are plotted in Fig. 2.

In Fig. 3, we show that the theoretical scaling $\mathcal{O}(N\log(N))$ can be achieved in practice. The crossover with MMM2D for a moderate accuracy of 10^{-4} in rms relative error of forces is found to be less than 20 particles. Both programs

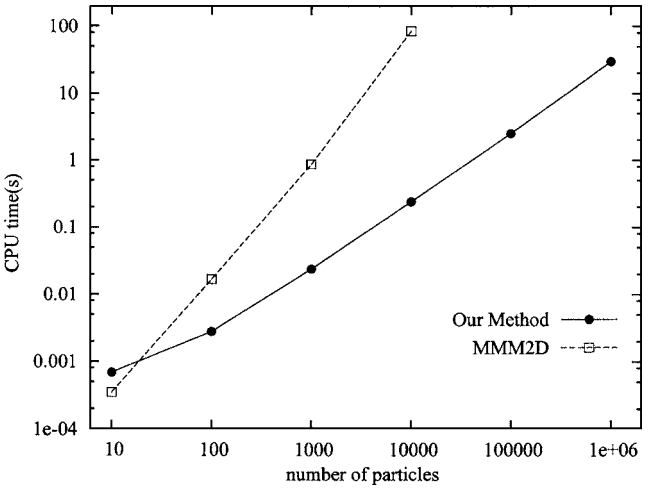

FIG. 3. CPU time of one time evaluation of forces on particles and potential energy with our method (solid curve) and MMM2D method (dashed curve).

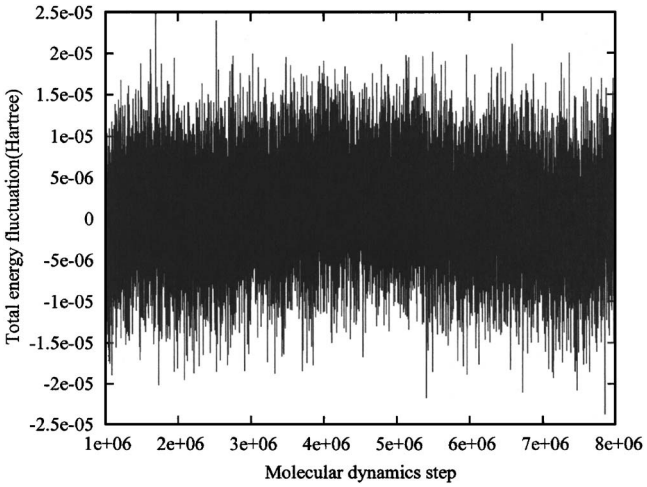

FIG. 4. The total energy fluctuations calculated with our method.

were run in AMD Opteron 2400 MHz. The degree of the finite elements is a parameter that can be optimized to obtain the smallest possible CPU time for a fixed accuracy. For high accuracies higher degrees are recommended. The CPU time for the calculation of the forces dominates in our method over the time needed to calculate the energy.

C. Energy conservation

Energy conservation is of great importance in molecular dynamics simulations. In order to investigate energy conservation in a real simulation, we performed a very long (8 ns) molecular dynamics simulation of a sodium chloride system containing 1000 particles. The velocity Verlet algorithm with a time step of 50 a.u. is used to update the particle positions and velocities. The short range interactions are obtained from the Born-Mayer-Huggins-Fumi-Tosi[26] rigid-ion potential, with the parameters of Ref. 27. The shortest oscillation period was of the order of 3000 atomic units, i.e., 60 molecular dynamics steps. After an equilibration for 1×10^6 steps, 7×10^6 steps were performed during which the total energy and potential energy were monitored. The fluctuation of the total energy, as shown in Fig. 4, has an oscillation amplitude of about 2.5×10^{-5}, while the amplitude of the potential energy oscillation was three orders of magnitude larger. The total energy was thus conserved very well.

D. The optimal method parameters

Since the short range part is similar to the known Ewald-type methods, the details concerning the error resulting from the cutting off of the summation in the real space can be found in Ref. 28. However, handling the error resulting from long range part is sophisticated because the long range part is solved by a method which is a mixture of plane wave and finite element. In addition, the new parameter, the degree of the polynomial for the finite element, plays an important role in connection with both accuracy and efficiency. This fact makes the choice of the method parameters an intricate task. To overcome this problem, we performed a large number of runs for a sodium chloride crystalline system including 1000 atoms, each run with different method parameters. Moreover,

TABLE I. The optimal method parameters obtained by the Pareto frontier optimization. All method parameters h_{xy}, h_z, $x_{\max}$, r_{cut}, and α are given in Å while average nearest-neighbor distance for sodium chloride system is 2.8 Å. The degree of the polynomial for the finite element is 7 for all the runs.

Rel. err.[a]	h_{xy}[b]	h_z[c]	$x_{\max}$[d]	r_{cut}[e]	α[f]	Time (s)
10^{-2}	1.55	2.33	4.0	5.00	2.10	0.012
10^{-3}	1.75	1.86	5.0	9.00	2.20	0.028
10^{-4}	1.55	1.64	7.0	10.0	2.10	0.040
10^{-5}	1.40	1.40	8.0	11.0	2.10	0.052
10^{-6}	1.27	1.33	7.5	12.0	2.20	0.068
10^{-7}	1.27	0.90	7.0	13.7	2.15	0.092
10^{-8}	1.07	0.70	7.9	13.7	2.10	0.120
10^{-9}	0.87	0.65	8.7	13.7	2.00	0.148
10^{-10}	0.93	0.56	8.7	14.0	1.95	0.172

[a]Ratio of rms force error to norm of forces.
[b]Grid spacing in the periodic directions.
[c]Grid spacing in the nonperiodic direction.
[d]Cutoff radius of the Gaussian charge distribution.
[e]Cutoff radius of the real space term.
[f]Ewald splitting parameter.

the optimal parameters are obtained from a Pareto frontier optimization. A point is on our Pareto frontier optimization if there is no other point which has both smaller CPU time and smaller ratio of rms force error to norm of forces [see Eq. (27)],

$$\sqrt{\frac{\sum_{i=1}^{N}(\mathbf{F}_i-\mathbf{F}_i^{\text{exact}})^2}{\sum_{i=1}^{N}(\mathbf{F}_i^{\text{exact}})^2}}. \tag{27}$$

All runs to determine the optimal method parameters are performed on an Intel(R) Pentium(R) 4 CPU 3.00 GHz. A set of optimal method parameters for a wide range of accuracy is given in the Table I. The parameter values given in Table I are optimal for the sodium chloride systems, and those are only rough values for other systems so that test simulations are recommended for other systems which their average nearest-neighbor distance is different from sodium chloride system.

IV. CONCLUSION

In this manuscript, we presented a method to solve Poisson's equation for smooth charge densities with periodic boundary condition in two directions and finite in the third one. It is very efficient for smooth charge densities and it does not require much memory. The resulting error distribution is uniform over the entire simulation cell. Our method is based on the plane wave representation in the periodic directions and finite elements in the nonperiodic direction. Based on this method, we can then calculate the electrostatic energy and forces of particles interacting by Coulombic potential with high accuracy and a $N\log(N)$ scaling. The method satisfies intrinsically and without any approximations the boundary conditions appropriate for surface problems. It is best suited for a moderate number of particles between 10^2 and 10^6. The method is expected to be suitable for an efficient parallelization since the time dominating parts are only loosely coupled.

ACKNOWLEDGMENTS

S. Alireza Ghasemi would like to thank M.J. Rayson for valuable and helpful discussions. This work has been supported by the Swiss National Science Foundation and the Swiss National Center of Competence in Research (NCCR) on Nanoscale Science.

APPENDIX: DETAILS FOR THE EXPLOITED FINITE ELEMENT METHOD

We consider a uniform grid on the interval $[z_l,z_u]$ with $N+1$ nodes $\{z_0,z_1,\ldots,z_N\}$, while $z_0=z_l$ and $z_N=z_u$. The interval is thus divided into N equally spaced subintervals (elements). The functions $d(z)$ and $c(z)$ are replaced by the approximate functions $D(z)$ and $C(z)$, which are expanded in the basis of Eq. (20) on each subinterval. We use the Galerkin approach in which the same bases are used for the expansion of both $D(z)$ and $C(z)$. Our bases are a combination of the hat function $\phi^v(z)$ centered at nodes

$$\phi_j^v(z)=\begin{cases}(z_{j+1}-z)/h, & z\in[z_j,z_{j+1})\\ (z-z_{j-1})/h, & z\in[z_{j-1},z_j)\\ 0, & \text{otherwise},\end{cases} \tag{A1}$$

and hierarchical polynomials[25] $\phi^m(z)$,

$$\phi_{j,i}^m(z)=\begin{cases}N_i[2(z-z_j)/h+1], & z\in[z_{j-1},z_j]\\ 0, & \text{otherwise},\end{cases} \tag{A2}$$

localized within the individual elements. N_i are given in canonical coordinates in Eq. (20). Finally, $C(z)$ and $D(z)$ within the element $[z_{j-1},z_j]$ will be

$$C(z)=c_{j-1}\phi_{j-1}^v(z)+c_j\phi_j^v(z)+\sum_{i=2}^{p}c_{j,i}\phi_{j,i}^m(z), \tag{A3a}$$

$$D(z)=d_{j-1}\phi_{j-1}^v(z)+d_j\phi_j^v(z)+\sum_{i=2}^{p}d_{j,i}\phi_{j,i}^m(z). \tag{A3b}$$

Note that because $\phi_{j,i}^m(z)$ vanishes at all nodes, we obtain $c_j = C(z_j)$. Replacing the approximate functions from Eqs. (A3a) and (A3b) in Eq. (25) gives

$$\sum_{j=1}^{N} [A_j(D,C) - (D,\eta)_j] = gd_0c_0 + gd_Nc_N. \tag{A4}$$

We split $A_j(D,C)$ as

$$A_j(D,C) = A_j^S(D,C) + A_j^M(D,C), \tag{A5}$$

where

$$A_j^S(D,C) := -\int_{z_{j-1}}^{z_j} D'(z)C'(z)dz, \tag{A6}$$

$$A_j^M(D,C) := -\int_{z_{j-1}}^{z_j} g^2 D(z)C(z)dz, \tag{A7}$$

$$(D,\eta)_j := \int_{z_{j-1}}^{z_j} D(z)\,\eta(z)dz. \tag{A8}$$

$C(z)$ within an element is

$$C(z) = \boldsymbol{\phi}_j^T(z)\mathbf{c}_j, \quad z \in [z_{j-1}, z_j], \tag{A9}$$

where $\mathbf{c}_j$ and $\boldsymbol{\phi}_j(z)$ are vectors with $p+1$ elements,

$$\mathbf{c}_j := [c_{j-1}, c_j, c_{j,2}, \ldots, c_{j,p}]^T, \tag{A10}$$

$$\boldsymbol{\phi}_j(z) := [\phi_{j-1}^v(z), \phi_j^v(z), \phi_{j,2}^m(z), \ldots, \phi_{j,p}^m(z)]^T. \tag{A11}$$

Then,

$$A_j^S(D,C) = \mathbf{d}_j^T K_j \mathbf{c}_j, \tag{A12}$$

$$A_j^M(D,C) = \mathbf{d}_j^T M_j \mathbf{c}_j, \tag{A13}$$

where

$$K_j := -\int_{z_{j-1}}^{z_j} \frac{d\boldsymbol{\phi}_j}{dz}\frac{d\boldsymbol{\phi}_j^T}{dz}, \tag{A14}$$

$$M_j := -\int_{z_{j-1}}^{z_j} g^2 \boldsymbol{\phi}_j \boldsymbol{\phi}_j^T. \tag{A15}$$

The $(p+1)\times(p+1)$ matrix K_j is called the element stiffness matrix and the $(p+1)\times(p+1)$ matrix M_j is called the element mass matrix. Although the element index j is present in the definitions of K_j and M_j, in our case of uniform grid spacing, these matrices do not depend on j. By performing the summations $\Sigma_{j=1}^N A_j^M$ and $\Sigma_{j=1}^N A_j^S$, we build up the global mass matrix and the global stiffness matrix. We arrange the order of elements of these matrices as

$$\mathbf{c} := \begin{bmatrix} \mathbf{c}_L \\ \mathbf{c}_Q \end{bmatrix}, \tag{A16}$$

$$\mathbf{c}_L := [c_0, c_1, \ldots, c_N]^T, \tag{A17}$$

$$\mathbf{c}_Q := [c_{1,2}, \ldots, c_{1,p}, \ldots, c_{N,2}, \ldots, c_{N,p}]^T, \tag{A18}$$

$$K = \begin{bmatrix} K_L & 0 \\ 0 & K_Q \end{bmatrix}, \tag{A19}$$

$$M = \begin{bmatrix} M_L & M_{LQ} \\ M_{LQ}^T & M_Q \end{bmatrix}. \tag{A20}$$

The second term of the summand in Eq. (A4) should be calculated approximately because only the values of $\eta(z)$ on the nodes are available,

$$(D,\eta)_j = \mathbf{d}_j^T \mathbf{I}_j, \tag{A21}$$

where

$$\mathbf{I}_j := \int_{z_{j-1}}^{z_j} \boldsymbol{\phi}_j(z)\,\eta(z)dz. \tag{A22}$$

Interpolating the integration is appropriate to calculate the above integral by fitting a polynomial of degree $d \geqslant 2p$ to the nodes of element $[z_{j-1}, z_j]$ and its neighboring nodes,

$$(\mathbf{I}_j)_i = \sum_{k=-p}^{p-1} w_k^i \eta_{j+k}. \tag{A23}$$

Recall that our charge density is localized within the interval $[z_l, z_u]$ and it smoothly tends to be zero at the edges. Therefore, it is appropriate to zero pad the ends of the $\eta(z)$. The coefficients w_k^i are weights from high-order interpolation. Building up the global matrices yields

$$(D,\eta) = \mathbf{d}^T\mathbf{I}, \tag{A24}$$

where the elements of the vector $\mathbf{d}$ are coefficients of expansion of test function $D(z)$ as denoted in Eq. (A3a) and the order of elements of $\mathbf{I}$ is the same as in Eq. (A16),

$$\mathbf{I} := \begin{bmatrix} \mathbf{I}_L \\ \mathbf{I}_Q \end{bmatrix}, \tag{A25}$$

$$\mathbf{I}_L := [I_0, I_1, \ldots, I_N]^T, \tag{A26}$$

$$\mathbf{I}_Q := [I_{1,2}, \ldots, I_{1,p}, \ldots, I_{N,2}, \ldots, I_{N,p}]^T. \tag{A27}$$

Finally, adding the right-hand side of Eq. (A4) to the global matrices yields

$$\begin{bmatrix} P_L & M_{LQ} \\ M_{LQ}^T & P_Q \end{bmatrix}\begin{bmatrix} \mathbf{c}_L \\ \mathbf{c}_Q \end{bmatrix} = \begin{bmatrix} \mathbf{I}_L \\ \mathbf{I}_Q \end{bmatrix}, \tag{A28}$$

where M_{LQ} is a sparse $(N+1)\times N(p-1)$ matrix,

$$P_Q := K_Q + M_Q \tag{A29}$$

is a $N(p-1)\times N(p-1)$ block-diagonal matrix,

$$P_L := K_L + M_L - ge_0e_0^T - ge_Ne_N^T \tag{A30}$$

is a tridiagonal $(N+1)\times(N+1)$ matrix, and

$$e_0 = [1, 0, \ldots, 0]^T, \tag{A31}$$

$$e_N = [0, \ldots, 0, 1]^T. \tag{A32}$$

Multiplying the matrix in Eq. (A28) and eliminating $\mathbf{c}_Q$ in the system of linear equations yield

$$[P_L - M_{LQ}P_Q^{-1}M_{LQ}^T]\mathbf{c}_L = \mathbf{I}_L - M_{LQ}P_Q^{-1}\mathbf{I}_Q. \tag{A33}$$

Finally, we obtain our system of linear equations,

$$B\mathbf{c}_L = \mathbf{b}, \tag{A34}$$

where the matrix B and the vector $\mathbf{b}$ are

$$B \coloneqq P_L - M_{LQ}P_Q^{-1}M_{LQ}^T, \tag{A35}$$

$$\mathbf{b} \coloneqq \mathbf{I}_L - M_{LQ}P_Q^{-1}\mathbf{I}_Q. \tag{A36}$$

It turns out that in the general case, the matrix B is symmetric tridiagonal of dimension $(N+1)\times(N+1)$. The proof for the tridiagonality of matrix B can be found in the context of block cyclic reduction.[29] Note that elements of the vector $\mathbf{c}_L$ are the values of $C(z)$ at the grid points. Therefore, by solving a system of linear equations, which has a tridiagonal matrix, we can find the values of $C(z)$ at the grid points. Instead of using the finite element method, we could have used finite differences to solve Eq. (11). Although calculating the right-hand-side $\mathbf{b}$ is computationally more expensive in our approach than in the finite difference method, the whole process of solving the system of linear equations is less expensive because the factorization of the tridiagonal matrix can be done fast.

[1] P. P. Ewald, Ann. Phys. **64**, 253 (1921).
[2] L. Greengard and V. Rokhlin, J. Comput. Phys. **73**, 325 (1987).
[3] R. W. Hockney and J. W. Eastwood, *Computer Simulation Using Particles* (Hilger, London, 1988).
[4] J. Lekner, Physica A **176**, 485 (1991).
[5] R. Sperb, Mol. Simul. **20**, 179 (1998).
[6] R. Strebel and R. Sperb, Mol. Simul. **27**, 61 (2001).
[7] M. Challacombe, C. White, and M. Head-Gordon, J. Chem. Phys. **107**, 10131 (1997).
[8] J. Hautman and M. L. Klein, Mol. Phys. **75**, 379 (1992).
[9] D. M. Heyes, M. Barber, and J. H. R. Clarke, J. Chem. Soc., Faraday Trans. 2 **73**, 1485 (1977).
[10] B. R. A. Nijboer, Physica A **125**, 275 (1984).
[11] A. H. Widmann and D. B. Adolf, Comput. Phys. Commun. **107**, 167 (1997).
[12] A. Arnold and C. Holm, Comput. Phys. Commun. **148**, 327 (2002).
[13] E. Spohr, J. Chem. Phys. **107**, 6342 (1997).
[14] I. C. Yeh and M. L. Berkowitz, J. Chem. Phys. **111**, 3155 (1999).
[15] Y. J. Rhee, J. W. Halley, J. Hautman, and A. Rahman, Phys. Rev. B **40**, 36 (1989).
[16] A. Arnold, J. de Joannis, and C. Holm, J. Chem. Phys. **117**, 2496 (2002).
[17] J. de Joannis, A. Arnold, and C. Holm, J. Chem. Phys. **117**, 2503 (2002).
[18] A. Bródka, Chem. Phys. Lett. **410**, 446 (2005).
[19] A. Neelov, S. A. Ghasemi, and S. Goedecker, J. Chem. Phys. **127**, 024109 (2007).
[20] E. L. Pollock and J. Glosli, Comput. Phys. Commun. **95**, 93 (1996).
[21] L. Genovese, T. Deutsch, and S. Goedecker, J. Chem. Phys. **127**, 054704 (2007).
[22] M. Frigo and S. G. Johnson, Proc. IEEE **93**, 216 (2005).
[23] J. E. Flaherty, course notes on finite element method at http://www.cs.rpi.edu/%7Eflaherje/
[24] S. Adjerid, M. Aiffa, and J. E. Flaherty, SIAM J. Appl. Math. **55**, 520 (1995).
[25] B. Szabó and I. Babuška, *Finite Element Analysis* (Wiley, New York, 1991).
[26] M. P. Tosi and F. G. Fumi, J. Phys. Chem. Solids **25**, 45 (1964).
[27] T. Zykova-Timan, D. Ceresoli, U. Targaglino, and E. Tosatti, J. Chem. Phys. **123**, 164701 (2005).
[28] J. Kolafa and J. W. Perram, Mol. Simul. **9**, 351 (1992).
[29] W. Gander and G. H. Golub, *Scientific Computing: Proceedings of the Workshop*, edited by G. H. Golub (Springer, New York, 1988).