Solving the Poisson equation for solute–solvent systems using fast Fourier transforms

Christine Peter, Wilfred F. van Gunsteren, and Philippe H. Hünenberger

Laboratorium für Physikalische Chemie, ETH Zürich, CH-8093 Zürich, Switzerland

(Received 17 January 2002; accepted 6 February 2002)

An iterative algorithm based on fast Fourier transforms is proposed to solve the Poisson equation for systems of heterogeneous permittivity (e.g., solute cavity in a solvent) under periodic boundary conditions. The method makes explicit use of the dipole–dipole interaction tensor, and is thus easily generalizable to arbitrary forms of electrostatic interactions (e.g., Coulomb’s law with straight or smooth cutoff truncation). The convergence properties of the algorithm and the influence of various model parameters are investigated in detail, and a set of appropriate values for these parameters is determined. The algorithm is further tested by application to three types of systems (a single spherical ion, two spherical ions, and small biomolecules), and comparison with analytical results (single ion) and with results obtained using a finite-difference solver under periodic boundary conditions. The proposed algorithm performs very well in terms of accuracy and convergence properties, with an overall speed comparable in the current implementation to that of a typical finite-difference solver. Future developments and applications of the algorithm will include: (i) the assessment of periodicity- and cutoff-induced artifacts in explicit-solvent simulations; (ii) the design of new electrostatic schemes for explicit-solvent simulations mimicking more accurately bulk solution; (iii) a faster evaluation of solvation free energies based on continuum electrostatics in cases where periodicity-induced artifacts can be neglected. © 2002 American Institute of Physics.

[DOI: 10.1063/1.1465396]

I. INTRODUCTION

When investigating the structure, dynamics, and function of (bio-)molecules through theoretical computational methods, the accurate description of solvation (and in particular of its electrostatic component) is an essential requirement to obtain reliable simulations.1–7 The methods available to handle solvation in molecular simulations can be sorted into two main categories, depending on whether they use an explicit or an implicit representation of the solvent molecules.

In explicit-solvent models,2,4,5 each solvent molecule is represented by a set of particles (usually atoms), interacting with each other and with the solute according to a given potential energy function (force field). In implicit-solvent models,1–4,7 the solvent molecules are not included in the simulation. Instead, their overall effect is mimicked by the inclusion of a mean-solvation term into the potential energy function for the solute (potential of mean force), and possibly the addition of stochastic and frictional forces8 acting on the solute atoms. The most popular implicit-solvent models are based on continuum electrostatics.9–12 In this approach, the solute is treated as a low-dielectric cavity encompassing the solute atomic point charges, and embedded into a dielectric continuum of permittivity equal to that of the solvent. The electrostatic potential in the system is computed by numerically solving the Poisson (or Poisson–Boltzmann, in the presence of implicit counterions) equation, giving access to the electrostatic solvation free energy of the solute.

As they ignore the discrete nature of the solvent, implicit-solvent models have some limitations. For models based on continuum electrostatics, one may mention the following shortcomings: (i) important sensitivity to empirical model parameters (atomic charges and radii, value of the solute dielectric permittivity, exact definition of the solute–solvent boundary); (ii) neglect of nonlinear effects (electrostriction, dielectric saturation); (iii) neglect of the solvent structure around the solute (structure in the first solvation shells, specific hydrogen bonds); (iv) difficulty to represent accurately the nonelectrostatic component of solvation (hydrophobic effect, solute–solvent van der Waals interactions). However, because the evaluation of the pairwise solvent–solvent interactions in explicit-solvent simulations is extremely time-consuming, the most attractive feature of implicit-solvent models is their reduced computational costs. In addition to being computationally expensive, explicit-solvent simulations have another major drawback; they are sensitive to the choice of boundary conditions (system size and shape, finite vs periodic system, method to handle electrostatic interactions).

The majority of explicit-solvent (bio-)molecular simulations are carried out under periodic boundary conditions,13,14 which has the advantage of removing any distortion associated with a solvent-vacuum boundary. In this case, electrostatic interactions can be handled in three ways: (i) by straight truncation of the Coulomb interactions at a convenient (cutoff) distance,13,14 (ii) by smooth truncation of the Coulomb interactions (e.g., using a switched or shifted interaction15–19 or including a reaction-field correction20–22);
(iii) by the use of lattice-sum methods (Ewald,\textsuperscript{23} P\textsuperscript{3}M,\textsuperscript{24} or PME, Ref. 25). Straight truncation leads to severe artifacts in many simulated properties of (bio-)molecular systems.\textsuperscript{26–36} Although the two latter choices presumably represent a less drastic approximation, artifacts (e.g., strong dependence of simulated observables on the cutoff distance or system size) have been evidenced in a number of cases for both smooth truncation methods\textsuperscript{19,32,37–39} and lattice-sum methods.\textsuperscript{22,40–43}

From these observations, the key problem appears to be the following: none of the currently available choices of boundary conditions for explicit-solvent simulations can make a simulated microscopic system (typically $<1000 \text{ nm}^3$) represent adequately an equivalent volume in the bulk of a solution.

If no choice of boundary conditions is adequate to mimic the bulk environment in explicit-solvent simulations, this is not the case in continuum-electrostatics (implicit-solvent) calculations. There, the boundary conditions to solve the Poisson equation are specified in the form of the potential at the surface of the computational box. For a reasonably large solute–wall distance, this potential is well approximated by the solvent-screened Coulomb potential of the solute charges.\textsuperscript{40} In addition, continuum-electrostatics methods have been recently generalized to other types of boundary conditions typically used in explicit-solvent simulations.\textsuperscript{34,35,40,44–47} By comparing the outcome of a continuum-electrostatics calculation under such a type of boundary conditions with that of another calculation under bulk boundary conditions, it is possible to investigate the nature and magnitude of boundary-condition artifacts in explicit-solvent simulations. This procedure is illustrated in Fig. 1 for the case of an explicit-solvent simulation employing the Ewald method (or a related mesh method) to compute electrostatic interactions.\textsuperscript{40–43}

The above approach based on continuum electrostatics represents a very general strategy to analyze and improve electrostatic schemes for explicit-solvent simulations, with the goal of understanding, correcting, and ultimately eliminating artifacts linked with inappropriate boundary conditions. However, its application requires the generalization of continuum electrostatics to the types of boundary conditions typical for explicit-solvent simulations. In the present article, we develop a method applicable to systems (i) under periodic boundary conditions, and (ii) involving an arbitrary form of electrostatic interactions, including no truncation (lattice-sum methods), straight truncation, or smooth truncation of the Coulomb interactions. Note that two previously developed models (based on finite-difference\textsuperscript{40} or fast Fourier transforms\textsuperscript{45}) satisfy the first requirement but not the second one, while a model developed for studying the solvation of spherical ions\textsuperscript{35} satisfies the second requirement but not the first one. To our knowledge, the only the algorithm cumulating both features is the dielectric self-consistent field method proposed by Boresch and Steinhauser.\textsuperscript{44}

With the goal of designing a general algorithm satisfying both conditions, we propose in the present article a new method to solve the Poisson equation using fast Fourier transforms (FFTs). This algorithm is designed for the following purposes: (i) the assessment of periodicity-induced artifacts in explicit-solvent simulations using the Ewald or related mesh methods;\textsuperscript{40–47} (ii) the design of new electrostatic schemes for explicit-solvent simulations, where a correction based on continuum electrostatics would enable proper bulk boundary conditions; (iii) the generalization of the algorithm to electrostatic interactions involving straight or smooth truncation of the Coulomb interactions for the assessment of cutoff-induced artifacts in explicit-solvent simulations; (iv) the faster evaluation of solvation free energies based on continuum electrostatics (FFT algorithms are highly optimized on many computers and benefit from an advantageous scal-
ing with the system size, as \( N \log N \) in cases where periodicity-induced artifacts can be neglected.

In the present article, the algorithm is described in details. The influence of various parameters controlling its behavior is investigated systematically, leading to a set of reasonable values to be used for further calculations. The behavior of the algorithm is further examined in the context of a solvated spherical ion, a system of two spherical ions, and four more complex biomolecular systems. The method is validated by comparison of the results with those of a finite system. Further applications of the algorithm to truncated electrostatic interactions will be presented in a future article.

II. THEORY

In this section, we describe a new method to solve the equations of continuum electrostatics for a periodic solute–solvent system by means of fast Fourier transforms (FFTs). This description is organized as follows. Section II A introduces the equations of continuum electrostatics and shows how, under periodic boundary conditions, they can be transformed into a tractable problem in reciprocal space. Section II B proposes an iterative method based on the use of FFTs to solve the problem set in Sec. II A. Section II C provides the equations for calculating the vacuum field in a periodic system of charges using the Ewald method. Finally, Sec. II D discusses some implementation issues related to the definition (and possible smoothing) of the solute–solvent dielectric boundary.

A. Continuum electrostatics in periodic systems

Using continuum electrostatics, a system consisting of a solute molecule immersed in a solvent is characterized by four position-dependent electrostatic quantities:\(^{26,27,48-50}\) the relative dielectric permittivity \( \varepsilon(\mathbf{r}) \), the vacuum electric field \( \mathbf{V}(\mathbf{r}) \), the polarization field \( \mathbf{P}(\mathbf{r}) \), and the electric field \( \mathbf{E}(\mathbf{r}) \). In the present discussion, we focus on an infinite periodic system constructed by replication of a (reference) rectangular unit cell of edges \( L_x \), \( L_y \), and \( L_z \) containing one (reference) solute molecule surrounded by the solvent (see Fig. 2). In this case, the four above quantities are periodic within the infinite system, and can be expanded into Fourier series with the coefficients \( \hat{\mathbf{V}}(\mathbf{k}) \), \( \hat{\mathbf{P}}(\mathbf{k}) \) and \( \hat{\mathbf{E}}(\mathbf{k}) \), where \( \mathbf{k} = 2\pi \mathbf{L}^{-1} \mathbf{l} \) with \( \mathbf{l} \in \mathbb{Z}^3 \), \( \mathbf{L} \) being a diagonal matrix with elements \( L_x \), \( L_y \), and \( L_z \).

Assuming a sharp solute–solvent dielectric boundary and a nonpolarizable solute (solute dielectric permittivity \( \varepsilon_s \) equal to one), the dielectric permittivity \( \varepsilon(\mathbf{r}) \) can be written

\[
\varepsilon(\mathbf{r}) = 1 + (\varepsilon_s - 1) H(\mathbf{r}),
\]

where \( \varepsilon_s \) is the relative permittivity of the solvent, and \( H(\mathbf{r}) \) is a function that evaluates to zero inside the reference solute molecule or any of its periodic copies, and to one otherwise. The vacuum field \( \mathbf{V}(\mathbf{r}) \) is defined as the electric field generated in the absence of solvent (\( \varepsilon_s = \varepsilon_r = 1 \)) by the atomic point charges of the reference solute molecule and all its periodic copies plus, in the case of a non-neutral solute, a homogeneous neutralizing background charge density filling the infinite periodic system.\(^{22,40,42,51,52}\) As detailed in Sec. II C, the vacuum field can be evaluated using the Ewald method.\(^{23}\) The polarization field \( \mathbf{P}(\mathbf{r}) \) represents the dipole moment density within the system, which results from the reorientation of the solvent dipoles in response to the local electric field. Taken together, the vacuum field and the electric (reaction) field generated by the polarized solvent yield the electric field \( \mathbf{E}(\mathbf{r}) \).

The two unknowns in such a continuum electrostatics problem, namely \( \mathbf{E}(\mathbf{r}) \) and \( \mathbf{P}(\mathbf{r}) \), are linked by two equations. First, the field equation\(^{26,27,49,50}\) states that the electric field is the sum of the vacuum field and the reaction field generated by the polarized solvent, namely,

\[
\mathbf{E}(\mathbf{r}) = \mathbf{V}(\mathbf{r}) + \int \int \int_{\mathbb{R}^3} d^3r' \, T(\mathbf{r}-\mathbf{r}') \mathbf{P}(\mathbf{r}'),
\]

where the integral is performed over the whole periodic system, and \( T(\mathbf{r}) \) is the dipole–dipole interaction tensor. This
tensor is defined in such a way that the field generated at \( \mathbf{r} \) by a point dipole \( \mathbf{\mu} \) at the origin is equal to \( \mathbf{T}(\mathbf{r})\mathbf{\mu} \). For Coulombic interactions, \( \mathbf{T}(\mathbf{r}) \) may be written \(^{26}\)

\[
\mathbf{T}(\mathbf{r}) = (4\pi\varepsilon_0)^{-1}\nabla \otimes \nabla \mathbf{r}^{-1} = (4\pi\varepsilon_0)^{-1}r^{-5}(3\mathbf{r} \otimes \mathbf{r} - r^2 \mathbf{1}),
\]

where \( \varepsilon_0 \) is the permittivity of vacuum, and the notation \( \mathbf{a} \otimes \mathbf{b} \) has been introduced for the tensor with the elements \( a_ib_j \). Second, the constitutive (or linear-response) equation\(^{26,27,48,50}\) states that the polarization in a dielectric continuum is proportional to the local electric field, namely,

\[
\mathbf{P}(\mathbf{r}) = \varepsilon_0(\varepsilon_s - 1)\mathbf{E}(\mathbf{r}) = \varepsilon_s(\varepsilon_s - 1)H(\mathbf{r})\mathbf{E}(\mathbf{r}).
\]

Equations (2) and (4) must be solved simultaneously for \( \mathbf{P}(\mathbf{r}) \) and \( \mathbf{E}(\mathbf{r}) \).

Because the system considered is periodic, one may attempt to express both equations in reciprocal space. This is easy for Eq. (2), which has the form of a convolution integral, leading to

\[
\tilde{\mathbf{E}}(\mathbf{k}) = \tilde{\mathbf{V}}(\mathbf{k}) + \tilde{\mathbf{T}}(\mathbf{k})\tilde{\mathbf{P}}(\mathbf{k}).
\]

As shown in Appendix A, the Fourier coefficients of the Coulombic dipole–dipole interaction tensor [Eq. (2)] are given by

\[
\tilde{\mathbf{T}}(\mathbf{k}) = -(3\varepsilon_0)^{-1}[3k^{-2}\mathbf{k} \otimes \mathbf{k}(1 - \delta_\mathbf{k}) + \mathbf{1}\delta_\mathbf{k}],
\]

where \( \delta_\mathbf{k} \) is the Kronecker delta (one if \( k = 0 \), zero otherwise). Unlike Eq. (5), the reciprocal-space counterpart of Eq. (4) is more complicated (a convolution) than Eq. (4) itself (a product). However, this difficulty could be avoided if the function \( H(\mathbf{r}) \) were removed from Eq. (4). The method proposed below to solve Eqs. (2) and (4) simultaneously in reciprocal space is based on this idea.

Let us define a modified vacuum field \( \mathbf{V}'(\mathbf{r}) \) which is equal to the vacuum field \( \mathbf{V}(\mathbf{r}) \) outside the reference solute molecule and its periodic copies, but possibly different inside. This condition may be written

\[
H(\mathbf{r})\mathbf{V}'(\mathbf{r}) = H(\mathbf{r})\mathbf{V}(\mathbf{r}).
\]

If we were to solve the electrostatic problem for the vacuum field \( \mathbf{V}'(\mathbf{r}) \) instead of \( \mathbf{V}(\mathbf{r}) \), the polarization would be unaffected (because the solute is non-polarizable), i.e.,

\[
\mathbf{P}'(\mathbf{r}) = \mathbf{P}(\mathbf{r}),
\]

while the field \( \mathbf{E}'(\mathbf{r}) \) would only differ from \( \mathbf{E}(\mathbf{r}) \) inside the solute, namely,

\[
\mathbf{E}'(\mathbf{r}) = \mathbf{E}(\mathbf{r}) + [1 - H(\mathbf{r})][\mathbf{V}'(\mathbf{r}) - \mathbf{V}(\mathbf{r})].
\]

By carefully choosing \( \mathbf{V}'(\mathbf{r}) \), it is possible to make \( \mathbf{E}'(\mathbf{r}) \) vanish inside the solute, i.e.,

\[
\exists \mathbf{V}'(\mathbf{r}) \quad \text{with} \quad \mathbf{E}'(\mathbf{r}) = H(\mathbf{r})\mathbf{E}'(\mathbf{r}).
\]

From here on, \( \mathbf{V}'(\mathbf{r}) \) is defined as the modified vacuum field for which Eq. (10) is satisfied. In this case, Eq. (4) can be rewritten in reciprocal space as

\[
\tilde{\mathbf{P}}(\mathbf{k}) = \varepsilon_0(\varepsilon_s - 1)\tilde{\mathbf{E}}'(\mathbf{k}),
\]

where \( \tilde{\mathbf{P}}(\mathbf{k}) \) was used instead of \( \tilde{\mathbf{P}}'(\mathbf{k}) \) due to Eq. (8). Equations (5) [for \( \tilde{\mathbf{E}}'(\mathbf{k}) \), \( \tilde{\mathbf{V}}'(\mathbf{k}) \) and \( \tilde{\mathbf{P}}(\mathbf{k}) \)] and (11) can now be combined and solved simultaneously, as (the last equality is proved in Appendix B),

\[
\tilde{\mathbf{E}}'(\mathbf{k}) = \tilde{\mathbf{V}}'(\mathbf{k}) + \varepsilon_0(\varepsilon_s - 1)\tilde{\mathbf{T}}(\mathbf{k})\tilde{\mathbf{E}}'(\mathbf{k})
\]

\[
= [1 - \varepsilon_s(\varepsilon_s - 1)\tilde{\mathbf{T}}(\mathbf{k})]^{-1}\tilde{\mathbf{V}}'(\mathbf{k})
\]

\[
= \left[ 1 + \frac{\varepsilon_0}{\varepsilon_s - 1} \tilde{\mathbf{T}}(\mathbf{k})(1 - \delta_\mathbf{k}) - \frac{\varepsilon_s - 1}{\varepsilon_s + 2}\delta_\mathbf{k} \right] \tilde{\mathbf{V}}'(\mathbf{k}).
\]

Using the above trick, the problem of solving the two-coupled equations has been transformed into that of finding the unique \( \tilde{\mathbf{V}}'(\mathbf{k}) \) for which \( \tilde{\mathbf{E}}'(\mathbf{k}) \), calculated via Eq. (12) followed by a Fourier transform, vanishes inside the solute as required by Eq. (10). An iterative procedure for finding this solution using FFTs is presented in Sec. II B.

After the determination of \( \tilde{\mathbf{V}}'(\mathbf{k}) \), the Fourier coefficients of the polarization field are obtained by application of Eqs. (11) and (12). The polarization can then be used to compute the reaction potential from the solvent and the electrostatic solvation free energy of the solute. In real space and for Coulombic interactions, the reaction potential \( \Phi_{RF}(\mathbf{r}) \) is related to the polarization through \(^{28}\)

\[
\Phi_{RF}(\mathbf{r}) = (4\pi\varepsilon_0)^{-1}\int \int \int_{\mathbb{R}^3} d^3\mathbf{r}' \frac{\mathbf{r} - \mathbf{r}'}{|\mathbf{r} - \mathbf{r}'|} \mathbf{P}(\mathbf{r}'),
\]

where the integral is performed over the whole periodic system. This expression is a convolution integral, and is easily transformed into its reciprocal-space counterpart (see Appendix A),

\[
\tilde{\Phi}_{RF}(\mathbf{k}) = -\varepsilon_0^{-1}k^{-2}\varepsilon_0^{-1}\mathbf{P}(\mathbf{k})(1 - \delta_\mathbf{k}).
\]

The electrostatic contribution to the solvation free energy is obtained by integrating the charge density (including the homogeneous neutralizing background charge for non-neutral systems) multiplied by the reaction potential and dividing by two, \(^{9,48}\) leading to

\[
\Delta G_{\text{solv}} = \frac{1}{2} \sum_i N_u q_i [\Phi_{RF}(\mathbf{r}_i) - \tilde{\Phi}_{RF}] = \frac{1}{2} \sum_i N_u q_i \tilde{\Phi}_{RF}(\mathbf{r}_i),
\]

where \( \mathbf{r}_i \) and \( q_i \) represent the coordinate vector and partial charge of atom \( i \) in the reference solute molecule, and \( \tilde{\Phi}_{RF} \) is the average of the reaction potential over the reference unit cell. The latter quantity vanishes due to the absence of \( k = 0 \) term in Eq. (14).

### B. Iteration procedure

The iterative procedure used in the present study to construct the modified vacuum field \( \mathbf{V}'(\mathbf{r}) \) for which \( \mathbf{E}'(\mathbf{r}) \) vanishes inside the solute [Eq. (10)] relies on the discretization of the rectangular reference unit cell by a grid \( G \) of \( N_x \cdot N_y \cdot N_z \) points, and the use of a three-dimensional (3D) fast Fourier transform (FFT) algorithm. In the following, the coordi-
nate vector of a grid point of \( G \) will be noted \( r_n = N^{-1}L_n \), where \( N \) is a diagonal matrix with elements \( N_x, N_y, \) and \( N_z \), and \( n \in G \) is a vector with integer components in the ranges \( [0 \cdots N_x - 1], \) \( [0 \cdots N_y - 1] \) and \( [0 \cdots N_z - 1] \). Similarly, \( k_l = 2 \pi L^{-1} \) \( l \) with \( l \in G \) will indicate the reciprocal-space vector corresponding to a grid point \( l \). The algorithm requires the definition of 6 grids of complex numbers to store the Cartesian components of the field pairs \( [V'(r_n); \hat{V}'(k_l)] \) and \([E'(r_n); \hat{E}'(k_l)]\), and consists of the following steps:

1. Compute the vacuum field \( V(r_n) \) on \( G \) using the Ewald method (see Sec. II C). Define the modified vacuum field at iteration \( i = 0 \) as
   \[
   V'_0(r_n) = H(r_n) V(r_n).
   \]
   Note that \( V'_0(r_n) \) has no imaginary component; (2) Apply three forward 3D-FFTs to obtain the Fourier coefficients \( V'_0(k_l) \) of \( V'_0(r_n); \)
2. Compute the residual \( \sigma_i \) of \( V'_0(r_n) \) inside the solute as
   \[
   \sigma_i = \frac{\sum_{n \in G} |V'_0(r_n)|^2 |1 - H(r_n)|^{1/2}}{\sum_{n \in G} |1 - H(r_n)|^{1/2}},
   \]
   where \( |\cdot| \) denotes the norm of a complex vector; (6a) If \( \sigma_i \) is above a predefined threshold \( \tau \) [i.e., Eq. (10) is not satisfied], choose the modified vacuum field for the next iteration according to
   \[
   V'_{i+1}(r_n) = V'_i(r_n) - \lambda [1 - H(r_n)] E'_i(r_n)
   \]
   and return to step (2). The parameter \( \lambda \) is introduced as a relaxation parameter. 53 Although \( \lambda = 1 \) is the obvious choice, overrelaxation (1 < \( \lambda < 2 \)) can significantly accelerate the convergence of the algorithm in practice (see Sec. IV B); (6b) If the residual \( \sigma_i \) of \( E'_i(r_n) \) is below the predefined threshold \( \tau \) [i.e., Eq. (10) is satisfied], terminate the iteration;
3. Apply three backward 3D-FFTs to obtain \( E'_0(r_n) \) from its Fourier coefficients \( E'_0(k_l) \); (7) Compute \( \Phi_{RF}(k_l) \) from \( E'_0(k_l) \) using Eqs. (11) and (14). Apply a backward 3D-FFT to obtain \( \Phi_{RF}(r_n) \) from its Fourier coefficients \( \Phi_{RF}(k_l) \). Compute the electrostatic solvation free energy using Eq. (15), where the reaction potential \( \Phi_{RF}(r) \) at the location of an atom \( i \) is estimated by linear interpolation of the values of \( \Phi_{RF}(r_n) \) at the eight surrounding grid points.

The above algorithm can be given a simple physical interpretation. Replacing Eq. (4) by Eq. (11) [i.e., replacing \( H(r) \) by 1] is effectively equivalent to assuming that the solute is polarizable, with a permittivity equal to that of the solvent. As a consequence, the only way to ensure that the polarization still vanishes within the solute is to remove the electric field inside the solute by addition of an offset to the vacuum field (i.e., use a modified vacuum field). A reasonable first guess is to set the modified vacuum field equal to zero inside the solute [step (1)]. However, the polarization of the solvent may generate a residual electric field (and thus a residual artifactual polarization) within the solute. These residual field and polarization must be removed by adapting the modified vacuum field [step (6a)]. However, this adaption changes again the field within the whole system, so that the procedure has to be repeated iteratively until self-consistency is achieved.

C. Computing the vacuum field

The vacuum field \( V(r) \) in a system under periodic boundary conditions can be computed through Ewald summation. 23,40,42 In this case, the vacuum potential generated by a point charge \( q \) at the origin, all its periodic copies, and a homogeneous neutralizing background charge density is written 40,42,46

\[
\Phi_{\text{vac}}(r) = \frac{q}{4 \pi \epsilon_0 \left| \sum_{n \neq \delta} \eta(\|r + L_n\|; a) \right|} + \frac{4 \pi}{V} \sum_{l \neq \delta} n^2 \frac{\gamma(\alpha \|k\| r)}{k^2} e^{i \alpha \|k\| r} + A_1.
\]

Here, \( \gamma(r;a) \) is a charge-shaping function, characterized by the width parameter \( a \) and satisfying the normalization condition,

\[
4 \pi \int_0^\infty dr r^2 \gamma(r;a) = 1.
\]

The \( \gamma(\alpha \|k\| r) \) terms represent the Fourier coefficients of (a lattice sum of) the charge-shaping function, and are given by

\[
\gamma(\alpha \|k\| r) = \begin{cases} 4 \pi \alpha^{-1} \int_0^\infty dr \sin(\alpha \|k\| r) \gamma(r;a) & \text{for } k \neq 0, \\ 1 & \text{for } k = 0. \end{cases}
\]

The function \( \eta(r;a) \) is related to \( \gamma(r;a) \) through

\[
\eta(r;a) = 4 \pi A r^{-1} \int_0^\infty dr' r'(r' - r) \gamma(r';a).
\]

Finally, the constant \( A_1 \) is given by

\[
A_1 = -\frac{4 \pi}{V} \int_0^\infty dr r^2 \eta(r;a).
\]

In the following, we only consider charge-shaping functions satisfying the property 46

\[
\gamma(r;a) = 0 \text{ for } r \gg a \text{ with } 2a \leq \min\{L_x, L_y, L_z\}.
\]

In this case, \( \eta(r;a) \) in Eq. (19) vanishes for any \( r \gg a \). Owing to this property, the field defined by

\[
W(r) = \sum_{\mathcal{V}} q_i w(r - r_i; a_i)
\]

with

\[
w(r,a) = -(\epsilon_0 V)^{-1} \sum_{l \neq \delta} \gamma(\alpha \|k\| r) k^2 e^{i \alpha \|k\| r}
\]

\[
= -(\epsilon_0 V)^{-1} \sum_{l \neq \delta} \frac{i k^2}{k^2} \gamma(\alpha \|k\| r)
\]

is identical to the vacuum field outside the reference solute molecule and its periodic copies, provided that for any atom
at a closer distance from any atom center than its radius $R_i$. We assume here that no point of the solute–solvent boundary lies at a closer distance from any atom center than its radius $R_i$.

A sufficient condition for this is the use of the van der Waals surface of the atoms as solute–solvent boundary, as will be the case in the present study. When the above condition is satisfied, one can thus write

$$H(r)V(r) = H(r)W(r).$$

(27)

As a consequence, knowledge of the quantity $W(r_n)$ is sufficient to initiate the iterative algorithm presented in Sec. II B [step (1); see Eq. (16)]. In practice, the procedure for evaluating $W(r_n)$ is the following:

1. Evaluate the reciprocal-space version of the fields $\tilde{w}(r_n; a_i)$ on the grid $G$ for the different values of $a_i$ in Eq. (25) as

$$\tilde{w}(k_i; a_i) = -\epsilon^{-1}_\infty \sum_{m \in Z^3} \tilde{\gamma}(a_i k_i + N_m) \frac{\epsilon^1(k_i + N_m)}{k_i + N_m} (1 - \delta_k).$$

(28)

The summation over the alias vectors $k_i + N_m = 2\pi L^{-1}(1 + Nm)$ of $k_i$ is required to convert the infinite Fourier series in Eq. (26) into a finite series. The convergence of Eq. (28) with the number of alias vectors (of increasing norm) included in the $m$-sum depends on the value of $a_i$. As a broad $\gamma$ function leads to a narrow $\tilde{\gamma}$ function, the best convergence will be obtained for the largest allowed value of $a_i$, which is in the present case the atomic radius $R_i$. However, evaluating the grids $\tilde{w}(k_i; R_i)$ for all possible atomic radii in the system may be computationally expensive. A better solution is to use a few distinct values of $a$, representing the maximal radius within a subset of atom types (e.g., one value of $a$ for hydrogen atoms and another value for all nonhydrogen atoms);

2. Evaluate the reciprocal-space version of the field $W(r_n)$ as [see Eq. (25)],

$$W(k_i) = \sum_{r_i} q_i e^{-ik_i \cdot \vec{r}} \tilde{w}(k_i; a_i);$$

(29)

3. Apply three backward 3D-FFTs to obtain $W(r_n)$ from its Fourier coefficients $W(k_i)$.

In the present study, we used two types of charge-shaping functions satisfying Eqs. (20) and (24):$^{46}$ the spherical hat function with

$$\gamma(r; a) = 3\pi^{-1} a^{-4}(a - r)h(a - r)$$

and

$$\tilde{\gamma}(ak) = 12(ak)^{-4}[2 - 2\cos(ak) - ak \sin(ak)],$$

(30)

and the spherical parabola function with

$$\gamma(r, a) = (15/2)\pi^{-1} a^{-5}(a - r)^2h(a - r)$$

and

$$\tilde{\gamma}(ak) = 60(ak)^{-5}[2ak + ak \cos(ak) - 3 \sin(ak)],$$

(31)

where $h(x)$ is the Heaviside function [i.e., $h(x) = 1$ for $x \geq 0$, $h(x) = 0$ otherwise]. In principle, the final result of the calculation should be independent of any specific choice of charge-shaping function and charge widths $a_i \leq R_i$. In practice, however, these choices may affect the accuracy of the algorithm when restricting the $m$-sum of Eq. (28) to the finite subset of alias vectors with integer components in the range $[-m_{\max} \leq m \leq m_{\max}]$ (see Sec. IV B).

D. Boundary smoothing

In the present method, some care must be taken with respect to the treatment of the solute-solvent dielectric boundary. All electrostatic quantities involved in the algorithm are discontinuous at this boundary, in particular the permittivity $\epsilon(r)$, the modified vacuum field $V(r)$, and the modified electric field $E'(r)$. Because sharp variations in the functions may lead to artifacts in their Fourier transforms, while the smoothing of these functions may affect the apparent size of the solute cavity, the exact definition of the boundary is an important parameter of the model. This definition must be specified at two different steps of the algorithm (see Sec. II B): (i) in step (1), when the vacuum field $V(r_n)$ inside the solute is set to zero to get $V(0)(r_n)$; (ii) in step (5) of each iteration, when evaluating the residual $\sigma(i)$, and in step (6a), when subtracting the residual field inside the solute from $V(i)(r_n)$ to obtain $V(i+1)(r_n)$. By experimenting with different choices of boundary definitions, it was found that applying boundary smoothing in the first case increased the accuracy of the method. However, in the second case, requiring the field $E'(r_n)$ to converge towards zero for points within the solute lying very close to the boundary led to slow convergence and increased errors. Such a layer of points with a nonvanishing electric field is probably needed to accurately represent the discontinuity in $E'(r)$ at the boundary.

To properly handle these two cases, two different “inside” functions $I(r_n)$ and $I'(r_n)$ were defined on the grid. The function $I(r_n)$ represents the volume fraction of the grid cell surrounding grid point $r_n$ that lies inside the solute. For points that are further away from the boundary than half a grid cell diagonal, $I(r_n)$ is either one (inside) or zero (outside). For the remaining grid points, $I(r_n)$ is determined as follows. The grid cell surrounding $r_n$ is divided into $(2n_{cub} + 1)^3$ cubelets. $I(r_n)$ is then evaluated as the fraction of the cubelets with their centers inside the solute. The function $I'(r_n)$ is equal to one whenever $I(r_n)$ is one, and to zero otherwise. The smoothed function $1 - I(r_n)$ is used in step (1) of the algorithm in place of $H(r_n)$; see Eq. (16). The sharp function $I'(r_n)$ is used in steps (5) and (6a) of the algorithm in place of $1 - H(r_n)$; see Eqs. (17) and (18).

III. COMPUTATIONAL DETAILS

To test our new algorithm, three types of periodic systems were considered, based on the following unit cells: (i) a single spherical ion in a cubic box; (ii) two spherical ions in a cubic box; and (iii) several small biomolecules. All computations relied on the FFT algorithms of the FFTW library.$^{57}$ The single-ion system is characterized by four physical parameters: the ionic charge $q$, the ionic radius $R$, the cubic box edge $L$, and the solvent permittivity $\epsilon_s$. For a limited set
of parameters \((q = \pm e; R = 0.2 \text{ nm}; L = 3.0 \text{ nm}; \epsilon_s = 2 \text{ or } 78)\), the convergence properties of the algorithm and the influence of model parameters were investigated in detail. The following model parameters were considered:

1. The grid resolution, determined by the number of grid points \(N_x = N_y = N_z = N\) discretizing the reference (cubic) unit cell along the three Cartesian axes;

2. The method used to compute the vacuum field (see Sec. II C), including the choice of a charge-shaping function \(\gamma\) and charge width \(a\), and the number of alias vectors included in the \(m\)-sum of Eq. (28). In practice, the latter sum is restricted to \(m\) vectors with integer components in the range \([-m_{\max}, m_{\max}]\);

3. The definition of the solute–solvent boundary (see Sec. II D), which relies on the choice of the discretization parameter \(n_{\text{cube}}\);

4. The relaxation parameter \(\lambda\) [see Sec. II B; step (6a) of the algorithm];

5. The threshold value \(\tau\) for the residual electric field inside the solute [see Sec. II B; step (6) of the algorithm].

For all parameters, an optimal value (or a range of appropriate values) was determined. All further calculations for the single- and the two-ion systems were performed using a spherical parabola charge-shaping function of width \(a = R - 0.05 \text{ nm}\), \(m_{\max} = 2\), \(n_{\text{cube}} = 4\), \(\lambda = 1.5\), and \(\tau = 0.3 \text{ kJ mol}^{-1} \text{ e}^{-1} \text{ nm}^{-1}\).

For the single-ion system and for different numbers of grid points \(N\), the influence of other parameters on the accuracy of the computed electrostatic solvation free energy \(\Delta G_{\text{solv}}\) was further investigated. These parameters included: (i) the location of the ion center relative to the grid (i.e., either at a grid point, or at the center of a grid cell defined by eight grid points); (ii) the dielectric permittivity of the solvent \(\epsilon_s\); (iii) the ionic radius \(R\). For this simple system, the accuracy of the algorithm could be evaluated by direct comparison of calculated \(\Delta G_{\text{solv}}\) with the corresponding analytical result: \(40\)

\[
\Delta G_{\text{soln}}^{\text{ana}} = -\frac{q^2}{4 \pi \epsilon_0} \frac{\epsilon_s - 1}{\epsilon_s} \frac{1}{2L} \times \left( \frac{\xi_{\text{EW}} + L}{R} + \frac{4\pi}{3} \left( \frac{R}{L} \right)^2 - \frac{16\pi^2}{45} \left( \frac{R}{L} \right)^5 \right),
\]

where \(\xi_{\text{EW}} \approx -2.837297\),\(^{51}\) The results of the FFT algorithm were also compared with those from a finite-difference (FD) algorithm. The FD solver used here is based on the UHBD program,\(^{54,58}\) adapted for periodic boundary conditions as described in Ref. 40.

The two-ion system is characterized by the following physical parameters: the ionic charges \(q_1\) and \(q_2\) (restricted here to \(q_1 = \pm q_2\)), the ionic radii \(R_1\) and \(R_2\) (restricted here to \(R_1 = R_2 = R\)), the interionic distance \(d\), the cubic box edge \(L\) and the solvent permittivity \(\epsilon_s\). For a limited set of parameters \((q_1 = \pm q_2 = 1 \text{ e}; R = 0.4 \text{ nm}; L = 3.0 \text{ nm})\), the influence of \(N, d\), and \(\epsilon_s\) on the computed electrostatic solvation free energies was examined, and the results compared with those of the FD solver. For another subset of parameters \((q_1 = \pm q_2 = 1 \text{ e}; R = 0.2 \text{ or } 0.4 \text{ nm}; L = 3.0 \text{ nm}; \epsilon_s = 78)\), the electrostatic solvation free energy was computed as a function of the interionic distance \(d\) and compared to the results of the FD solver.

For the biomolecular systems, the partial charges \(q_i\) of the atoms were assigned according to the GROMOS96 force field\(^{59,60}\) (for the peptide, as reported in Ref. 61). The corresponding atomic radii were calculated from the GROMOS Lennard-Jones \(C_6\) and \(C_{12}\) parameters for the interaction of each atom with an SPC water\(^{62}\) oxygen, as \(R = (2C_1/C_6)^{1/6}\). This expression defines the atom–water distance at the minimum of the Lennard-Jones curve. The computation of the vacuum field as described in Sec. II C requires that all atoms have a nonzero radius. Therefore, hydrogen atoms were treated in a special way and assigned a radius of 0.12 nm. The use of a relatively large value ensures that hydrogen atoms pointing into the solvent always contain at least a few grid points in their inside (for reasonably fine grids) and thus are not treated as part of the solvent. Since it is computationally inefficient to evaluate the vacuum field for many different charge widths [see Sec. II C; point (1)], the atoms were divided into two groups, with two different choices for the parameters \(a\) and \(m_{\max}\). The first group contained all hydrogen atoms, with \(a_1 = 0.06 \text{ nm} \text{ and } m_{\max,1} = 4\). The second group contained all nonhydrogen atoms, with \(a_2 = 0.24 \text{ nm} \text{ and } m_{\max,2} = 2\). In both cases, the selected value of \(a\) is smaller by at least 0.06 nm than the smallest radius of any atom in the group. The remaining model parameters were chosen as follows: \(n_{\text{cube}} = 4\), \(\lambda = 1.5\), and \(\tau = 0.3 \text{ kJ mol}^{-1} \text{ e}^{-1} \text{ nm}^{-1}\). For four biomolecular systems (a small aminooxy-acid tetrapeptide;\(^{61}\) the protein hen egg white lysozyme in the presence or absence of nine chloride counter ions; the protein \(\alpha\)-lactalbumin) in cubic periodic boxes (peptide: \(L = 6 \text{ nm}\); proteins: \(L = 8 \text{ nm}\)), the electrostatic solvation free energies computed using the FFT- and FD-solvers were compared for three different grid sizes \((N = 120, 140, \text{ or } 160)\). The configurations used in these calculations were taken from explicit solvent MD simulations reported elsewhere.\(^{61,63,64}\)

IV. RESULTS

A. Convergence properties

The convergence properties of the algorithm are illustrated in Figs. 3 and 4 for the simple case of a spherical ion \((q = \pm 1 \text{ e}; R = 0.2 \text{ nm})\) surrounded by a solvent of low or high permittivity \((\epsilon_s = 2 \text{ or } 78)\) in a cubic periodic box \((L = 3.0 \text{ nm})\). The evolution of the modified electric field \(E'(r_n)\) and of the reaction potential \(\Phi_{\text{Rf}}(r_n)\) during the iteration process are illustrated in Fig. 3 for \(\epsilon_s = 78\). The behavior of the modified electric field is consistent with the physical interpretation of the algorithm proposed at the end of Sec. II B. At iteration zero, the polarized solvent outside the ion induces a nonvanishing \(E'(0)(r_n)\) within the ion. By iterating, this component is progressively removed. The removal is essentially achieved after 100 iterations. At this point, the shape of \(E'(100)(r_n)\) is roughly similar to that of \(V'(0)(r_n)\), but its magnitude is significantly smaller (by about a factor \(\epsilon_s^{-1}\) at a distance larger than 0.05 nm from the ion.
FIG. 3. Evolution of the real component of the modified electric field \( E_i(r) \) along the \( x \)-axis (panel a; after iterations \( i = 0, 1, 10, \) and 100), and of the reaction potential \( \Phi_{RF}(r) \) (panel c; solid line, after iteration 0; dotted line, after iteration 100) during the iteration process. The imaginary component of the modified electric field (not displayed) is small (after iteration zero, it oscillates between about \( \pm 2 \) \( \text{kJ mol}^{-1} \text{e}^{-1} \text{nm}^{-1} \) homogeneously over the unit cell) and vanishes upon iterating. The component of the modified vacuum field at step 0, \( V_{0i}(r) \), along the \( x \)-axis is also shown for comparison (panel b; this function is real). All functions are computed for a single ion in the center of a cubic box \( (q = +1e; R = 0.2 \text{ nm}; L = 3.0 \text{ nm}; \epsilon_s = 78; N = 80; \gamma \) hat function; \( a = 0.15 \text{ nm}; m_{\text{max}} = 2; n_{\text{cube}} = 4; \lambda = 1.5) \), and displayed as a function of the coordinate \( x \) along a line parallel to the \( x \)-axis passing through the ion center.

FIG. 4. Convergence behavior of the FFT algorithm for a single ion at the center of a cubic periodic box \( (q = +1e; R = 0.2 \text{ nm}; L = 3.0 \text{ nm}; \gamma \) parabola function; \( a = 0.15 \text{ nm}; m_{\text{max}} = 2; n_{\text{cube}} = 4; \lambda = 1.5) \) in a solvent of low \( (\epsilon_s = 2; \text{upper panels}) \) or high \( (\epsilon_s = 78; \text{lower panels}) \) permittivity. The evolution of the residual \( \sigma_{(i)} \) of the electric field inside the solute (panels a and d) and of the electrostatic solvation free energy \( \Delta G_{\text{solv}} \) (panels b, c, e, and f) are displayed as a function of the number of iterations \( i \). For \( \Delta G_{\text{solv}} \), the ion center is located either at a grid point (panels b and e) or at a grid-cell center (panels c and f). For \( \sigma_{(i)} \), only the result for the ion centered at a grid point is displayed (the curves for the ion at a grid-cell center are very similar). The different curves correspond to different numbers of grid points \( (60 < N < 160) \), and the arrows indicate the analytical values \( \Delta G_{\text{solv}}^{\text{ana}} \) from Eq. (32).
surface; closer to the ion, the scaling factor is smaller). In contrast, the overall shape of the reaction potential and its magnitude at the ion center are not significantly altered by the iteration process.

The evolution of the residual $\sigma_i$ of the electric field inside the solute and of the electrostatic solvation free energy $\Delta G_{\text{solv}}$ is shown in Fig. 4 as a function of the number of iterations for several grid-spacings and for two different values of the solvent permittivity ($\varepsilon_s = 2$ or 78). To estimate the order of magnitude of the grid discretization errors, the convergence properties of $\Delta G_{\text{solv}}$ are displayed for the ion center located at either a grid point or a grid-cell center. The convergence of both the residual and the solvation free energy are significantly faster for the lower value of $\varepsilon_s$. For both values of the solvent permittivity, the solvation free energy displays much smaller relative variations upon iterating (i.e., it converges faster) compared to the residual. This observation could have been anticipated from Fig. 3, where large changes in the modified electric field within the ion during the iteration process were not correlated with large changes in the reaction potential. For all further calculations, the value $\tau = 0.3$ kJ mol$^{-1}$ e$^{-1}$ nm$^{-1}$ was selected as a convergence criterion on $\sigma_i$ to terminate the algorithm. Based on Fig. 4, this choice should lead to $\Delta G_{\text{solv}}$ values within at most 2% of their values for $i \rightarrow \infty$.

Although the converged solvation free energies calculated using an increasingly large number $N$ of grid points tend to be more accurate (compared to the analytical result), this trend is not systematic. For example, when the ion center is located at a grid point, the accuracy increases in the order $N = 60, 120, 100, 80, 160$, and 140 for $\varepsilon_s = 78$, and in the order $N = 100, 80, 160, 60, 120$, and 140 for $\varepsilon_s = 2$. However, in all cases, the converged numerical estimates are always within 3% of the analytical value (and within 0.5% for $N \geq 140$). The irregularity in the variation of the accuracy with the grid resolution is probably related to the fact that the number of grid points inside the solute and in the boundary region [where $\mathbf{I}(\mathbf{r}_a) \neq \mathbf{I}'(\mathbf{r}_a)$; see Sec. II D] is not a simple function of the number of grid points $N$ along each Cartesian axis. This interpretation is consistent with the observation that the order of accuracy as a function of $N$ changes upon shifting the ion center from a grid point to a grid-cell center. For similar reasons, in calculations involving larger ions (data not shown), the convergence of the residual becomes faster and discrepancies between numerical and analytical solvation free energies are reduced.

### B. Influence of model parameters

Various combinations of model parameters ($N$, $\gamma$, $a$, $m_{\text{max}}$, and $n_{\text{cube}}$) were tested for the case of a simple spherical ion ($q = +1$ e; $R = 0.2$ nm) in a cubic periodic box ($L = 3.0$ nm) filled by a solvent of permittivity $\varepsilon_s = 78$. The results are shown in Table I. In the upper part of the table, the influence of $\gamma$, $a$, and $m_{\text{max}}$ is investigated for $n_{\text{cube}} = 4$ and $N = 60$ or 80. For $a = 0.15$ nm or 0.10 nm, two main observations can be made. First, for a given value of $N$, $\Delta G_{\text{solv}}$ converges towards a common value for all choices of $\gamma$ and $a$ upon increasing the number of alias vectors. The most significant change in $\Delta G_{\text{solv}}$ occurs between $m_{\text{max}} = 0$ and 1, i.e., including at least the 26 nonzero alias vectors with components $-1, 0$ or $+1$ is essential for the accuracy of the algorithm. Second, the convergence with increasing $m_{\text{max}}$ is slower for $a = 0.10$ nm compared to $a = 0.15$ nm. This is expected because the alias sum in Eq. (28) converges faster when the Fourier coefficients of the charge-shaping function decrease rapidly with $k$, i.e., when $\gamma$ is a broad function (see Sec. II C). For $a = 0.20$ nm, $\Delta G_{\text{solv}}$ does not converge to the

---

**Table I. Influence of model parameters on the calculated electrostatic solvation free energy $\Delta G_{\text{solv}}$ for a single ion in a cubic periodic box ($q = +1$ e; $R = 0.2$ nm; $L = 3.0$ nm; $\varepsilon_s = 78$; $\lambda = 1.5$; $\tau = 0.3$ kJ mol$^{-1}$ e$^{-1}$ nm$^{-1}$).** The corresponding analytical value $\Delta G_{\text{solv}}^{\text{ana}}$ [see Eq. (32)] is 278.45 kJ mol$^{-1}$. The following model parameters are considered: $\gamma$, charge-shaping function; $a$, width of the charge-shaping function; $m_{\text{max}}$, maximal absolute value of the components of the alias vectors used for computing the vacuum field [see Eq. (28)]; $n_{\text{cube}}$, boundary-smoothing parameter (see Sec. II D). Calculated values are reported for either $N = 60$ or 80 grid points along each Cartesian axis.

<table>
<thead>
<tr>
<th>$m_{\text{max}}$</th>
<th>$n_{\text{cube}}$</th>
<th>$\gamma$</th>
<th>$a = 0.20$ nm</th>
<th>$a = 0.15$ nm</th>
<th>$a = 0.10$ nm</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>4</td>
<td>hat</td>
<td>$-270.73$</td>
<td>$-279.66$</td>
<td>$-263.22$</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>hat</td>
<td>$-271.21$</td>
<td>$-279.55$</td>
<td>$-271.24$</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>hat</td>
<td>$-271.23$</td>
<td>$-279.56$</td>
<td>$-271.29$</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>hat</td>
<td>$-271.24$</td>
<td>$-279.56$</td>
<td>$-271.30$</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>parabola</td>
<td>$-272.18$</td>
<td>$-280.03$</td>
<td>$-289.68$</td>
</tr>
<tr>
<td>5</td>
<td>4</td>
<td>parabola</td>
<td>$-271.34$</td>
<td>$-279.60$</td>
<td>$-279.79$</td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>parabola</td>
<td>$-271.31$</td>
<td>$-279.59$</td>
<td>$-279.63$</td>
</tr>
<tr>
<td>7</td>
<td>4</td>
<td>parabola</td>
<td>$-271.31$</td>
<td>$-279.58$</td>
<td>$-279.60$</td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>parabola</td>
<td>$-271.31$</td>
<td>$-279.58$</td>
<td>$-279.59$</td>
</tr>
<tr>
<td>9</td>
<td>0</td>
<td>parabola</td>
<td>$-305.01$</td>
<td>$-304.87$</td>
<td>$-304.90$</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>parabola</td>
<td>$-269.73$</td>
<td>$-274.16$</td>
<td>$-269.94$</td>
</tr>
<tr>
<td>11</td>
<td>2</td>
<td>parabola</td>
<td>$-267.76$</td>
<td>$-279.25$</td>
<td>$-267.97$</td>
</tr>
<tr>
<td>12</td>
<td>4</td>
<td>parabola</td>
<td>$-271.31$</td>
<td>$-279.59$</td>
<td>$-279.63$</td>
</tr>
<tr>
<td>13</td>
<td>6</td>
<td>parabola</td>
<td>$-271.21$</td>
<td>$-281.21$</td>
<td>$-281.26$</td>
</tr>
<tr>
<td>14</td>
<td>8</td>
<td>parabola</td>
<td>$-271.14$</td>
<td>$-280.81$</td>
<td>$-280.86$</td>
</tr>
</tbody>
</table>
same results as for the smaller values of $a$. In addition, the solvation free energies computed using the hat and parabola charge-shaping functions differ significantly, even for the largest value of $m_{\text{max}}$. These discrepancies can be explained as follows. The inside function $I'(r_n)$ used to compute the residual and update the modified vacuum field at each iteration (see Sec. II D) is set to zero for all grid points that are either outside the solute, or inside the solute and within $\Delta r = \sqrt{3L/2N}$ (half the length of a grid-cell diagonal) of the solute surface. As a consequence, the modified vacuum field in this boundary layer just below the solute surface is at all iterations equal to $V'_{(0)}(r_n)$. For this reason, the vacuum field for points in this layer should be computed exactly, in the sense that it should not contain information about the charge-shaping function (see Sec. II C). This means that $a$ should not only be smaller than the atomic radius $R$, but actually smaller than $R - \Delta r$. For $L=3.0$ nm, $R=0.2$ nm, and $N=60$ (or 80), this condition is satisfied for $a \approx 0.157$ nm (or 0.168 nm), i.e., only for the two smallest values of $a$ in Table I. Based on the above observations, further calculations (see Secs. IV C–IV D) were performed using a parabola charge-shaping function of width $a = R - 0.05$ nm (adequate for $N=60$ or larger when $L=3.0$ nm) and $m_{\text{max}}=2$. In principle, the calculation of the vacuum field scales with $m_{\text{max}}^3$, so that the use of large values of $m_{\text{max}}$ is computationally expensive. However, this parameter is only used in the calculation of the field grids $w(r_n; a_i)$ in Eq. (28). Since these grids do not contain any information about the molecular system considered, they can be precomputed once for all and read from a file for the actual calculation.

In the lower part of Table I, the effect of increasing $n_{\text{cube}}$, i.e., performing a more accurate smoothing at the solute–solvent boundary when computing $V'_{(0)}(r_n)$, is investigated for a parabola charge-shaping function with $m_{\text{max}} = 2$. Here again, the largest change is observed upon increasing $n_{\text{cube}}$ from 0 to 1. For $n_{\text{cube}}=4$, the variations become rather small (about 1 kJ mol$^{-1}$, i.e., <0.5%), so that all further calculations (see Secs. IV C–IV F) were performed with $n_{\text{cube}}=4$. The computational cost of the boundary-smoothing method employed here increases with $n_{\text{cube}}^3$, which makes this procedure rather inefficient. However, this method of smoothing does not introduce any analytical approximation, and remains tractable because it only needs to be applied to a small subset of grid points.

The influence of the relaxation parameter $\lambda$ on the number of iterations needed to reach convergence, i.e., to reach $\sigma_{(i)} \leq \tau$ with $\tau = 0.3$ kJ mol$^{-1}$ e$^{-1}$ nm$^{-1}$ (see Sec. IV A), is shown in Fig. 5. As already observed in Fig. 4, the convergence becomes systematically slower upon increasing the value of $\epsilon_s$. For all values of the solvent permittivity, the algorithm only converges for $\lambda$-values smaller than 2. However, the convergence can be accelerated significantly by using a $\lambda$-value larger than one (overrelaxation). The optimal value is always between 1 and 2, and depends on the actual value of $\epsilon_s$ (see Fig. 5), as well as on the number of grid points $N$ (data not shown). For further calculations, an intermediate $\lambda$-value of 1.5 was selected.

C. Results for a spherical ion

The calculated electrostatic solvation free energies $\Delta G_{\text{solv}}$ for a single spherical ion ($q=+1e; R=0.2$ or 0.4 nm) in a cubic periodic box ($L=3.0$ nm) filled by a continuum solvent ($\epsilon_s=2$ or 78) are reported in Table II. For numbers of grid points $N$ ranging from 60 to 160, five different estimates of $\Delta G_{\text{solv}}$ are compared, computed using the following methods: (i) present algorithm, with the ion center located at a grid point (FFT); (ii) present algorithm, with the ion center located at a grid-cell center (FFT*); (iii) present algorithm, with the ion center located at a grid point, but terminating the algorithm after iteration zero (FFT0); (iv) finite-difference solver, with the ion center located at a grid point (FD); (v) finite-difference solver, with the ion center located at a grid-cell center (FD*). These estimates can be compared with the analytical result $\Delta G_{\text{solv}}^{\text{ana}}$ from Eq. (32). Upon increasing the number of grid points, the FFT algorithm rapidly (but not always systematically) converges towards the analytical value of the solvation free energy. For
TABLE II. Electrostatic solvation free energy $\Delta G_{\text{solv}}$ of a single spherical ion of radius $R$ in a cubic periodic box ($q = +1e$; $L = 3.0$ nm; $\gamma$: parabola function; $a = R - 0.05$ nm; $m_{\text{max}} = 2$; $n_{\text{cub}} = 4$; $\lambda = 1.5$; $\sigma = 0.3$ kJ mol$^{-1}$ e$^{-1}$ nm$^{-1}$) filled with a solvent of permittivity $\epsilon_s$. Values are reported for different numbers of grid points $N$ along the three Cartesian axes, and the analytical estimate $\Delta G_{\text{solv}}^\text{ana}$ [see Eq. (32)] is given for comparison. The following methods are compared: FFT, present algorithm, ion center located at a grid point; FFT*, as FFT, but with the ion center located at a grid-cell center; FFT$^0$, as FFT, but performing only iteration zero of the algorithm; FD, finite-difference solver, ion center located at a grid point; FD*, as FD, but with the ion center located at a grid-cell center.

<table>
<thead>
<tr>
<th>$R$ (nm)</th>
<th>$\epsilon_s$</th>
<th>Method</th>
<th>$N = 60$</th>
<th>$N = 80$</th>
<th>$N = 100$</th>
<th>$N = 120$</th>
<th>$N = 140$</th>
<th>$N = 160$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.2</td>
<td>2</td>
<td>FFT</td>
<td>$-140.83$</td>
<td>$-141.64$</td>
<td>$-142.65$</td>
<td>$-141.09$</td>
<td>$-141.03$</td>
<td>$-141.59$</td>
</tr>
<tr>
<td>0.2</td>
<td>2</td>
<td>FFT*</td>
<td>$-142.54$</td>
<td>$-141.80$</td>
<td>$-141.49$</td>
<td>$-141.30$</td>
<td>$-141.25$</td>
<td>$-141.20$</td>
</tr>
<tr>
<td>0.2</td>
<td>2</td>
<td>FFT$^0$</td>
<td>$-141.24$</td>
<td>$-141.74$</td>
<td>$-142.69$</td>
<td>$-141.26$</td>
<td>$-141.14$</td>
<td>$-141.66$</td>
</tr>
<tr>
<td>0.2</td>
<td>2</td>
<td>FD</td>
<td>$-145.43$</td>
<td>$-143.36$</td>
<td>$-142.26$</td>
<td>$-141.77$</td>
<td>$-141.46$</td>
<td>$-140.69$</td>
</tr>
<tr>
<td>0.2</td>
<td>2</td>
<td>FD*</td>
<td>$-144.60$</td>
<td>$-142.63$</td>
<td>$-141.96$</td>
<td>$-142.11$</td>
<td>$-141.79$</td>
<td>$-141.85$</td>
</tr>
<tr>
<td>0.2</td>
<td>78</td>
<td>FFT</td>
<td>$-271.34$</td>
<td>$-279.59$</td>
<td>$-279.76$</td>
<td>$-274.96$</td>
<td>$-278.05$</td>
<td>$-278.94$</td>
</tr>
<tr>
<td>0.2</td>
<td>78</td>
<td>FFT*</td>
<td>$-279.38$</td>
<td>$-275.29$</td>
<td>$-276.26$</td>
<td>$-274.41$</td>
<td>$-278.84$</td>
<td>$-277.86$</td>
</tr>
<tr>
<td>0.2</td>
<td>78</td>
<td>FFT$^0$</td>
<td>$-278.87$</td>
<td>$-279.84$</td>
<td>$-281.72$</td>
<td>$-278.90$</td>
<td>$-279.67$</td>
<td>$-279.69$</td>
</tr>
<tr>
<td>0.2</td>
<td>78</td>
<td>FD</td>
<td>$-292.89$</td>
<td>$-289.60$</td>
<td>$-284.68$</td>
<td>$-281.73$</td>
<td>$-281.72$</td>
<td>$-280.13$</td>
</tr>
<tr>
<td>0.2</td>
<td>78</td>
<td>FD*</td>
<td>$-294.64$</td>
<td>$-286.19$</td>
<td>$-283.91$</td>
<td>$-283.61$</td>
<td>$-282.65$</td>
<td>$-282.61$</td>
</tr>
<tr>
<td>0.4</td>
<td>2</td>
<td>FFT</td>
<td>$-54.88$</td>
<td>$-55.11$</td>
<td>$-54.83$</td>
<td>$-54.85$</td>
<td>$-54.92$</td>
<td>$-54.83$</td>
</tr>
<tr>
<td>0.4</td>
<td>2</td>
<td>FFT*</td>
<td>$-54.98$</td>
<td>$-54.93$</td>
<td>$-54.89$</td>
<td>$-54.86$</td>
<td>$-54.86$</td>
<td>$-54.85$</td>
</tr>
<tr>
<td>0.4</td>
<td>2</td>
<td>FFT$^0$</td>
<td>$-54.96$</td>
<td>$-55.15$</td>
<td>$-54.87$</td>
<td>$-54.88$</td>
<td>$-54.94$</td>
<td>$-54.85$</td>
</tr>
<tr>
<td>0.4</td>
<td>2</td>
<td>FD</td>
<td>$-55.21$</td>
<td>$-54.68$</td>
<td>$-54.75$</td>
<td>$-55.08$</td>
<td>$-55.02$</td>
<td>$-54.94$</td>
</tr>
<tr>
<td>0.4</td>
<td>2</td>
<td>FD*</td>
<td>$-55.38$</td>
<td>$-55.25$</td>
<td>$-54.98$</td>
<td>$-55.07$</td>
<td>$-54.96$</td>
<td>$-54.93$</td>
</tr>
<tr>
<td>0.4</td>
<td>78</td>
<td>FFT</td>
<td>$-106.85$</td>
<td>$-108.51$</td>
<td>$-107.96$</td>
<td>$-107.88$</td>
<td>$-108.25$</td>
<td>$-108.02$</td>
</tr>
<tr>
<td>0.4</td>
<td>78</td>
<td>FFT$^0$</td>
<td>$-108.51$</td>
<td>$-108.88$</td>
<td>$-108.33$</td>
<td>$-108.36$</td>
<td>$-108.48$</td>
<td>$-108.30$</td>
</tr>
<tr>
<td>0.4</td>
<td>78</td>
<td>FD*</td>
<td>$-110.82$</td>
<td>$-110.33$</td>
<td>$-109.42$</td>
<td>$-109.58$</td>
<td>$-109.16$</td>
<td>$-109.02$</td>
</tr>
</tbody>
</table>

For $N > 100$ the numerical estimate of $\Delta G_{\text{solv}}$ is always within 1% of $\Delta G_{\text{solv}}^\text{ana}$. As observed previously, the convergence is faster for larger ions (larger number of grid points inside the ion and within the boundary layer at the ion surface; see Sec. IV A) and for lower values of the dielectric permittivity (smaller magnitude of the field discontinuity at the dielectric boundary). The difference between the two estimates of $\Delta G_{\text{solv}}$ with the ion center at a grid point or at a grid-cell center generally decreases in magnitude with increasing $N$. For $N > 100$, this difference is at most 0.8% of $\Delta G_{\text{solv}}^\text{ana}$. This quantity can serve as a simple measure of the discretization errors in the calculation. Note, however, that the two estimates do not systematically bracket the analytical result, and that the sign of their difference may vary with $N$.

The convergence properties of the FFT algorithm with $N$ are slightly superior to those of the FD solver. Especially for the coarser grids, the FD solver tends to overestimate solvation free energies, which is probably due to the boundary smoothing method used in this algorithm. This observation should be kept in mind when interpreting the small differences between results computed using the FFT and FD algorithms for more complex systems (Secs. IV D–IV F).

Finally, it appears that the solvation free energies computed from the polarization at iteration zero of the FFT algorithm are already very close to the corresponding converged estimates (see also Fig. 4). In some cases, these nonconverged estimates may even be closer to the analytical result than the converged ones. If this feature was general, suppressing the iteration process could lead to significant savings of computer time. Unfortunately, as will be shown in the next sections, this observation is specific to the case of a single spherical ion. For more complex systems, the iteration process is absolutely necessary to obtain accurate solvation free energies.

D. Results for two spherical ions

Here, we consider a system of two spherical ions ($q_1 = \pm q_2 = 1e$; $R = 0.4$ nm) in a cubic periodic box ($L = 3.0$ nm). The reaction potential $\Phi_{RF}(r)$ in a plane passing through both ion centers and running parallel to the $xy$-plane is shown in Fig. 6 for two ions with charges of opposite sign. This figure illustrates the smooth evolution of $\Phi_{RF}(r)$ upon varying the interionic distance from a separated ion-pair to ions at contact distance and to overlapping ions (diatomic molecule).

Results for the electrostatic solvation free energy are shown in Table III for two ions ($R = 0.4$ nm) of opposite charges ($q_1 = \pm q_2 = 1e$) at three interionic distances ($d = 0.2, 0.8, \text{and } 1.2$ nm), and for two ions of identical charges ($q_1 = q_2 = 1e$) at $d = 0.8$ nm, this for two values of the solvent permittivity ($\epsilon_s = 2$ or 78). As in Table II, solvation free energies are computed through three variants of the FFT algorithm (center-of-mass of the ions at a grid point, center-of-mass at a grid-cell center, and values after iteration zero) and two variants of the FD algorithm (center-of-mass at a grid point or at a grid-cell center). For the finer grids, the converged values computed using the FFT and FD algorithms
agree well, the difference being at most 2% for $N=160$. As already observed in the single-ion case (see Sec. IV C), values computed using the FD solver tend to be larger in magnitude compared to those based on the FFT solver. However, since the results based on the FFT algorithm were more accurate in the single-ion case, this discrepancy may result from a systematic error in the FD algorithm (probably related to boundary smoothing).

Shifting the center-of-mass of the ions from a grid point to a grid-cell center has little effect on the results of the FFT algorithm, especially for the finest grids, the relative difference between the two estimates being at most 0.4% for $N>100$. The effect of the shifting on the results of the FD solver is comparable, the relative difference being at most 0.3% for $N>100$. Here again, the difference between these two values may serve as a crude estimate for discretization errors. Finally, in contrast to the single-ion case, the solvation free energies computed from the polarization at iteration zero of the FFT algorithm differ noticeably from the corresponding converged values. For $N=160$, the relative difference between converged and nonconverged estimates increases with decreasing interionic distance (e.g., for opposite charges and $\epsilon_r=78$: 37% for $d=0.2$ nm; 4.9% for $d=0.8$ nm; 0.2% for $d=1.2$ nm). Thus, the solvation free energy at iteration zero appears to be a reasonable estimate only for isolated or widely separated ions. In the general case, however, iterating is necessary to obtain accurate solvation free energies.

The solvation free energy as a function of the interionic distance (i.e., the contribution of electrostatic solvation to the potential of mean force between the ions) is displayed in Fig. 7 for two ions of equal or opposite charges, computed either with the FFT or with the FD algorithm. The following observations can be made: (i) the curves are periodic of period $L$ and symmetrical with respect to $d=L/2$, i.e., the values of $\Delta G_{\text{solv}}$ for the interionic distances $d$ and $L-d$ are identical (only a half-period of the function is displayed in Fig. 7); (ii) the mean solvent force (negative derivative of the curves in Fig. 7) is attractive for identical charges and repulsive for opposite charges, i.e., this force opposes the direct Coulombic force between the ions; (iii) at $d=0$, $\Delta G_{\text{solv}}$ evaluates to zero for opposite charges (uncharged cavity) or to four times $\Delta G_{\text{solv}}$ evaluated for a single ion of charge $1e$ [see Eq. (32)] for equal charges (single ionic cavity with the double charge); (iv) at $d=L/2$, $\Delta G_{\text{solv}}$ evaluates to about twice $\Delta G_{\text{solv}}$ for a single ion (nearly noninteracting ions); (v) the derivative of $\Delta G_{\text{solv}}$ as a function of $d$ is zero at $d=L/2$ due to periodicity and symmetry (i.e., there is no solvation-induced force in this configuration). These observations are consistent with those reported in an earlier study of ion pairs based on the FD algorithm. The agreement between the results obtained by the FFT and FD algorithms is good. As discussed previously, the FD algorithm tends to slightly overestimate the magnitude of $\Delta G_{\text{solv}}$.

**E. Results of biomolecules**

As a final test for the stability and accuracy of the present FFT algorithm, electrostatic solvation free energies were computed for four, more complex, biomolecular systems. The four systems considered were (see Table IV): (i) a neutral aminoxy-acid tetrapeptide; (ii) the protein hen egg white lysozyme (HEWL, net charge $+9e$) together with 9 Cl$^-$ counterions; (iii) the same protein without counterions; and (iv) the protein $\alpha$-lactalbumin (net charge $-6e$). Here also, the calculations using the FFT and the FD solvers were performed with the center-of-mass of the protein either at a grid point or at a grid-cell center, so as to estimate the magnitude of discretization errors. The resulting electrostatic solvation free energies are reported in Table IV. The agreement between the results from the two solvers is very good, the deviation being at most 2.5% for $N=160$. As observed previously (see Secs. IV C–IV D) the solvation free energies computed using the FD solver are systematically larger in magnitude (except for the peptide system). The effect of shifting the system center-of-mass from a grid point to a grid-cell center on the results for both algorithms is fairly
The following methods are compared: FFT, present algorithm, center-of-mass (COM) of the two ions located at a grid point; FFT*, as FFT, but with the COM located at a grid-cell center; FFT0, as FFT, but performing only iteration zero of the algorithm; FD, finite-difference solver, COM located at a grid point; FD*, as FD, but with the COM located at a grid-cell center.

F. Computational efficiency

In order to compare the computational efficiencies of the FFT and FD algorithms, we considered the HEWL system without counterions (see Table IV). In Fig. 8, the convergence of $\Delta G_{\text{solv}}$ is displayed as a function of number of iteration cycles (starting with iteration 0 for both algorithms), for three choices of number of grid points ($N = 120$, 140, and 160). For all values of $N$, the FFT solver approaches the final result considerably faster than the FD solver. The FFT algorithm converges within 1% of the final $\Delta G_{\text{solv}}$ after 19 iterations for $N = 120$ (10 iterations for $N = 160$). In contrast, for the FD solver, convergence within 1% is only reached after 33 iterations for $N = 120$ (48 iterations for $N = 160$). The corresponding numbers for convergence within 5% of the final $\Delta G_{\text{solv}}$ are 4 (FFT, $N = 120$), 3 (FFT, $N = 160$), 23 (FD, $N = 120$), and 32 (FD, $N = 160$). Note also, that the speed of convergence of the FFT algorithm is nearly invariant with the number of grid points $N$, whereas this speed significantly decreases with increasing $N$ for the FD solver.

Since the two solvers are programmed in different lan-
guages (C for he FFT solver, Fortran77 for the FD solver) and optimized to different levels, it is difficult to perform a very accurate comparison in terms of computational efficiency. As a rough estimate, on a PC with a 1.2 GHz AMD processor and for $N=120$, one FFT iteration takes 4.1 s versus 0.7 s for one FD iteration. Since the FD algorithm requires about six times more iterations for convergence within 5% of the final result, the overall efficiencies of the two algorithms appear to be roughly comparable.

V. CONCLUSIONS

In the present study, we developed an iterative algorithm based on fast Fourier transforms (FFTs) to solve the Poisson equation for solute-solvent systems under periodic boundary conditions (see Fig. 2). The convergence properties of the algorithm and the influence of various model parameters (charge-shaping function $\gamma$, set of charge widths $a_i$, maximal components of alias vectors $m_{\text{max}}$, boundary-smoothing parameter $n_{\text{cube}}$, relaxation parameter $\lambda$, and residual convergence threshold $\tau$) were investigated in detail. After determination of a set of appropriate values for these different parameters, the algorithm was further tested by application to three types of systems: a single spherical ion, two spherical ions, and small biomolecules. The accuracy of the algorithm was assessed by comparison with the results obtained using a finite-difference (FD) solver$^{6,65}$ under periodic boundary conditions.

TABLE IV. Electrostatic solvation free energy $\Delta G_{\text{solv}}$ for biomolecules in cubic periodic boxes of edge $L$ filled with water ($\varepsilon=78$; $\gamma$ parabola function; $a_1=0.06$ nm; $a_2=0.24$ nm; $m_{\text{max}} = 4$; $m_{\text{max}} = 2$; $n_{\text{cube}} = 4$; $\lambda = 1.5$; $\tau = 0.3$ kJ mol$^{-1}$ nm$^{-1}$). Values are reported for different numbers of grid points $N$ along the three Cartesian axes. The following methods are compared: FFT, present algorithm, solute center-of-mass (COM) located at a grid point; FFT*, as FFT, but with the COM located at a grid-cell center; FD, finite-difference solver, COM located at a grid point; FD*, as FD, but with the COM located at a grid-cell center.

<table>
<thead>
<tr>
<th>System</th>
<th>$N$</th>
<th>FFT</th>
<th>FFT*</th>
<th>FD</th>
<th>FD*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tetrapeptide (49 atoms)</td>
<td>120</td>
<td>-8.50</td>
<td>-8.17</td>
<td>-8.14</td>
<td>-8.04</td>
</tr>
<tr>
<td>Net charge: 0e</td>
<td>140</td>
<td>-8.13</td>
<td>-8.13</td>
<td>-8.07</td>
<td>-8.06</td>
</tr>
<tr>
<td>$L=6.0$ nm</td>
<td>160</td>
<td>-8.11</td>
<td>-8.02</td>
<td>-7.93</td>
<td>-7.98</td>
</tr>
<tr>
<td>HEWL + $9\text{Cl}^-$ (1331 atoms)</td>
<td>120</td>
<td>-2518</td>
<td>-2518</td>
<td>-2578</td>
<td>-2579</td>
</tr>
<tr>
<td>Net charge: 0e</td>
<td>140</td>
<td>-2514</td>
<td>-2522</td>
<td>-2567</td>
<td>-2569</td>
</tr>
<tr>
<td>$L=8.0$ nm</td>
<td>160</td>
<td>-2506</td>
<td>-2511</td>
<td>-2559</td>
<td>-2558</td>
</tr>
<tr>
<td>HEWL (1322 atoms)</td>
<td>120</td>
<td>-2276</td>
<td>-2275</td>
<td>-2287</td>
<td>-2286</td>
</tr>
<tr>
<td>Net charge: $9e$</td>
<td>140</td>
<td>-2261</td>
<td>-2269</td>
<td>-2279</td>
<td>-2282</td>
</tr>
<tr>
<td>$L=8.0$ nm</td>
<td>160</td>
<td>-2256</td>
<td>-2260</td>
<td>-2275</td>
<td>-2274</td>
</tr>
<tr>
<td>$\alpha$-lactalbumin (1244 atoms)</td>
<td>120</td>
<td>-2183</td>
<td>-2180</td>
<td>-2221</td>
<td>-2218</td>
</tr>
<tr>
<td>Net charge: $-6e$</td>
<td>140</td>
<td>-2178</td>
<td>-2175</td>
<td>-2213</td>
<td>-2210</td>
</tr>
<tr>
<td>$L=8.0$ nm</td>
<td>160</td>
<td>-2174</td>
<td>-2172</td>
<td>-2204</td>
<td>-2206</td>
</tr>
</tbody>
</table>
conditions, and with the analytical result in the case of a single ion. An estimate of discretization errors was obtained by using different numbers of grid points $N$, and by comparing the results of two calculations with the solute translated by half a grid-cell diagonal relative to the grid. The following general observations were made:

(a) The algorithm converges faster for lower values of the solvent permittivity $\varepsilon_s$. This is to be expected since the magnitude of the discontinuity in the field and polarization at the solute–solvent boundary increases with $\varepsilon_s$.

(b) In the special case of a single spherical ion (or two spherical ions separated by a large distance), the solvation free energy is not significantly altered by the iteration procedure, i.e., the estimate after iteration zero is very close to the value after convergence of the residual to zero. This is no longer the case for a solute cavity containing a dipole (e.g., two ions at close distances, biomolecules). This observation can be understood qualitatively considering the interpretation of the algorithm proposed at the end of Sec. II B. For a single spherical ion, the solvent polarization after iteration zero is (almost) spherically symmetric, and gives rise to a very small residual field within the ion. On the other hand, for a dipolar solute, the solvent polarization after iteration zero generates a larger (dominantly dipolar) residual field within the solute. As a consequence, the removal of the residual field within the solute requires fewer iterations in the former case than in the latter case.

(c) The computed solvation free energies generally become more accurate upon increasing the number of grid points $N$. The difference between the two free-energy estimates involving the solute center translated by half a grid-cell diagonal relative to the grid also tends to decrease in magnitude with increasing $N$. However, the evolution of both quantities with $N$ is not completely systematic, and an increase in $N$ does not always result in a gain of accuracy. This nonsystematic behavior is probably due to the fact that the number of grid points within the solute (and in the boundary layer under the solute surface) does not depend in a simple fashion on the grid resolution and on the location of the solute relative to the grid.

(d) For a given number of grid points $N$, the accuracy of the FFT algorithm is generally slightly better (as judged by comparison with the analytical value for a single spherical ion) than the accuracy of the FD solver. The latter algorithm almost systematically overestimates the magnitude of solvation free energies, which may be a consequence of the method employed for boundary smoothing.

(e) For the biomolecular systems and for a (typical) number of grid points of 120, the FFT solver requires about six times fewer iterations than the FD solver to reach convergence of the solvation free energy within 5%. However, each iteration of the FFT algorithm is about six times more expensive in terms of computer time. Consequently, the overall speeds of the two algorithms are roughly comparable.

Although these results are very satisfactory in terms of accuracy, the FFT solver in its current implementation is not faster than a typical FD solver. Thus, a first goal for future improvements will be to enhance the speed of the computation. For this purpose, the following possibilities are currently being investigated: (i) the computation of the vacuum field may be accelerated by the use of the P$^3$M method instead of the Ewald method; (ii) the dielectric boundary smoothing may be accelerated by replacing the cubelits method by a computationally more efficient method; (iii) because any imaginary component in real-space observables is an artifact, the complex-to-complex FFT routines could be replaced by faster complex-to-real and real-to-complex FFT routines; (iv) the algorithm may be advantageously reformulated using the displacement field $D(r)$ in-

![FIG. 8. Convergence properties of the electrostatic solvation free energy $\Delta G_{\text{sol}}$ as a function of the number of iterations, for HEWL without counter ions ($L=8.0$ nm; $\varepsilon_s = 78$; $N=120$, 140 or 160; \gamma, parabola function: $a_1 = 0.06$; $a_2 = 0.24$; $m_{\text{max},1} = 4$; $m_{\text{max},2} = 2$; $n_{\text{max}} = 4$; $\lambda = 1.5$; $\tau = 0.3 \text{ kJ mol}^{-1} \text{ e}^{-1} \text{nm}^{-1}$). Results are reported for the FFT and the FD algorithm.](image-url)
stead of the electric field \( \mathbf{E}(\mathbf{r}) \), because the former quantity is not discontinuous at the solute–solvent boundary; (v) the algorithm may also be reformulated using the electrostatic potential \( \Phi(\mathbf{r}) \) instead of the electric field, with the goal of replacing the three three-dimensional FFTs per step (the same effort is required in similar FFT-based algorithms\(^{44,45} \)) by two one-dimensional FFTs.

A second goal for future development in view of practical applications is the generalization of the FFT algorithm to polarizable solutes (i.e., characterized by a solute permittivity differing from one). Finally, the major advantage of the present formalism resides in its explicit use of the dipole–dipole interaction tensor. For this reason, the third direction of future development will be the generalization of the method to other forms of dipole–dipole interaction tensor corresponding to electrostatic interactions typical of explicit-solvent simulations under periodic boundary conditions, namely straight or smooth truncation of the Coulomb interactions (work in progress). This should permit the semiquantitative study of the nature and magnitude of cutoff artifacts in explicit-solvent molecular simulations, and possibly the design of correction terms for the observables computed from these simulations. In the long run, it is our hope that the better understanding of periodicity- and cutoff-induced artifacts provided by continuum electrostatics will trigger and simplify the development of more accurate boundary conditions for explicit-solvent simulations aiming at reproducing bulk solution properties.

**ACKNOWLEDGMENT**

Financial support from the Swiss National Science Foundation (Project Nos. 21-50929.97 and 21000-61939) is gratefully acknowledged.

**APPENDIX A: FOURIER COEFFICIENTS OF THE DIPOLE–DIPOLE INTERACTION TENSOR**

Here, we derive the expression for the Fourier coefficients of the Coulombic dipole–dipole interaction tensor [Eq. (6)]. An equation relating the Fourier coefficients of the reaction field to those of the polarization field [Eq. (14)] is also derived as a by-product.

The Fourier coefficients of the interaction tensor describing the field generated by a periodic array of point dipoles can be written as

\[
\hat{T}(\mathbf{k}) = \int \int d^3 \mathbf{r} \exp(-i\mathbf{k} \cdot \mathbf{r}) \sum_{n \in \mathbb{Z}^3} T(\mathbf{r} + \mathbf{L} n)
\]

\[
= \int \int d^3 \mathbf{r} \exp(-i\mathbf{k} \cdot \mathbf{r}) T(\mathbf{r}), \quad (A1)
\]

where the first integral over the volume of the reference unit cell could be replaced by an integral over the whole space owing to periodicity.\(^{22} \) The cases \( k = 0 \) and \( k \neq 0 \) will be treated separately.

For \( k = 0 \), inserting Eq. (3) into Eq. (A1) leads to

\[
\hat{T}(0) = (4 \pi \epsilon_0)^{-1} \int \int d^3 \mathbf{r} \nabla \otimes \nabla \mathbf{r}^{-1}
\]

\[
= - (4 \pi \epsilon_0)^{-1} \lim_{\rho \to \infty} \rho^{-2} \int \Sigma(\rho) d^2 \sigma \mathbf{r} \otimes \mathbf{r}^{-1}
\]

\[
= - (4 \epsilon_0)^{-1} \int_0^\pi d\theta \sin(\theta) \mathbf{X}(\theta) = - (3 \epsilon_0)^{-1} \mathbf{1}, \quad (A2)
\]

where \( \Sigma(\rho) \) is the surface of a spherical volume \( \Omega(\rho) \) of radius \( \rho \) centered at the origin, and

\[
\mathbf{X}(\theta) = \pi^{-1} \int_0^{2\pi} d\phi \frac{r}{r} \otimes \frac{r}{r}
\]

\[
= \begin{pmatrix}
\sin^2(\theta) & 0 & 0 \\
0 & \sin^2(\theta) & 0 \\
0 & 0 & 2 \cos^2(\theta)
\end{pmatrix}. \quad (A3)
\]

The second equality in Eq. (A2) follows from applying the gradient theorem in \( \Omega(\rho) \) and inserting \( \nabla \mathbf{r}^{-1} = -\mathbf{r}^{-3} \mathbf{r} \), and the third equality from expressing the integrand in spherical coordinates and integrating over \( \phi \). This result was derived previously in the context of the calculation of the permittivity of liquids from molecular simulations.\(^{26,27,50} \)

For \( k \neq 0 \), one has to evaluate

\[
\hat{T}(\mathbf{k}) = (4 \pi \epsilon_0)^{-1} \int \int d^3 \mathbf{r} \exp(-i\mathbf{k} \cdot \mathbf{r}) \nabla \otimes \nabla \mathbf{r}^{-1}. \quad (A4)
\]

The integrand can be rewritten

\[
e^{-i\mathbf{k} \cdot \mathbf{r}} \mathbf{r} \otimes \mathbf{r}^{-1}
\]

\[
= \nabla \otimes (e^{-i\mathbf{k} \cdot \mathbf{r}} \nabla \mathbf{r}^{-1}) + i\mathbf{k} \otimes (e^{-i\mathbf{k} \cdot \mathbf{r}} \nabla \mathbf{r}^{-1})
\]

\[
= \nabla \otimes (e^{-i\mathbf{k} \cdot \mathbf{r}} \nabla \mathbf{r}^{-1}) + i\mathbf{k} \otimes (e^{-i\mathbf{k} \cdot \mathbf{r}} \nabla \mathbf{r}^{-1})
\]

\[
= -\mathbf{k} \otimes e^{-i\mathbf{k} \cdot \mathbf{r}} \mathbf{r}^{-1}. \quad (A5)
\]

Inserting this expression into Eq. (A4), applying the gradient theorem in a spherical volume \( \Omega(\rho) \) of arbitrarily large radius, and using \( \nabla \mathbf{r}^{-1} = -\mathbf{r}^{-3} \mathbf{r} \) leads to

\[
\hat{T}(\mathbf{k}) = (4 \pi \epsilon_0)^{-1} \lim_{\rho \to \infty} \rho^{-2} \int \Sigma(\rho) d^2 \sigma e^{-i\mathbf{k} \cdot \mathbf{r}} \mathbf{r} \otimes \mathbf{r}^{-1}
\]

\[
+ i\mathbf{k} \otimes \rho^{-2} \int \Sigma(\rho) d^2 \sigma e^{-i\mathbf{k} \cdot \mathbf{r}} \mathbf{r} \otimes \mathbf{r}^{-1}
\]

\[
- \mathbf{k} \otimes \lim_{\rho \to \infty} \left[ \int \Omega(\rho) d^3 \mathbf{r} \exp(-i\mathbf{k} \cdot \mathbf{r}) \mathbf{r}^{-1} \right]
\]

\[
= (4 \pi \epsilon_0)^{-1} \lim_{\rho \to \infty} [\mathbf{A}(\mathbf{k};\rho) + i\mathbf{k} \otimes \mathbf{b}(\mathbf{k};\rho) + \mathbf{k} \otimes \mathbf{c}(\mathbf{k};\rho)]. \quad (A6)
\]

The three quantities \( \mathbf{A} \), \( \mathbf{b} \), and \( \mathbf{c} \) are calculated separately below. For the tensor \( \mathbf{A} \), one has
\[
\mathbf{A}(\mathbf{k}; \rho) = -\rho^{-2} \int \int_{\Omega(\rho)} d^2 \sigma \ e^{-i \mathbf{k} \cdot \mathbf{r}} \frac{\mathbf{r}}{r} \otimes \frac{\mathbf{r}}{r}
\]
\[
= -\pi \int_0^\pi d\theta \sin(\theta) \cos[k \rho \cos(\theta)] \mathbf{X}(\theta)
\]
\[
= -\pi \int_{-1}^1 d\tau \cos(k \rho \tau) \begin{pmatrix}
1 - \tau^2 & 0 & 0 \\
0 & 1 - \tau^2 & 0 \\
0 & 0 & 2 \tau^2
\end{pmatrix}.
\] (A7)

The second equality results from expressing the integrand in spherical coordinates with the \(z\)-axis along \(\mathbf{k}\), observing that the imaginary component vanishes due to the symmetry of \(\mathbf{r} \otimes \mathbf{r}\), and integrating over \(\phi\). The last equality follows from the change of variable \(\tau = \cos(\theta)\). Substituting
\[
\int_{-1}^1 d\tau \cos(k \rho \tau) = \frac{2}{k \rho} \sin(k \rho)
\] (A8)
and
\[
\int_{-1}^1 d\tau \tau^2 \cos(k \rho \tau) = \frac{2}{k^2 \rho^2} \sin(k \rho) + \frac{4}{k^2 \rho^2} \cos(k \rho)
\]
\[- \frac{4}{k^2 \rho^2} \sin(k \rho)
\] (A9)
into Eq. (A7) permits the evaluation of the tensor \(\mathbf{A}\). It is then easily seen that \(\mathbf{A}\) vanishes in the limit \(\rho \to \infty\).

For the vector \(\mathbf{b}\), one has
\[
\mathbf{b}(\mathbf{k}; \rho) = \rho^{-1} \int \int_{\Omega(\rho)} d^2 \sigma \ e^{-i \mathbf{k} \cdot \mathbf{r}} \frac{\mathbf{r}}{r}
\]
\[
= -2 \pi ik^{-1} \mathbf{k} \int_0^\pi d\theta \sin(\theta) \cos(\theta) \sin[k \rho \cos(\theta)]
\]
\[
= -2 \pi ik^{-1} \mathbf{k} \int_{-1}^1 d\tau \tau \sin(k \rho \tau)
\]
\[
= 4 \pi k^{-2} \left[ \cos(k \rho) - \frac{1}{k \rho} \sin(k \rho) \right].
\] (A10)
The second equality results from using a spherical coordinate system with the \(z\)-axis along \(\mathbf{k}\), observing that the real component vanishes due to the antisymmetry of \(\mathbf{r}\), and integrating over \(\phi\). Since the \(x\)- and \(y\)-components vanish after this integration, the resulting vector is parallel to \(\mathbf{k}\).

For the scalar \(c\), one has
\[
c(\mathbf{k}; \rho) = -\int \int_{\Omega(\rho)} d^3 r \ e^{-i \mathbf{k} \cdot \mathbf{r}} r^{-1}
\]
\[
= -2 \pi \int_0^\rho d\rho \int_0^\pi d\theta \cos(\theta) \sin[k \rho \cos(\theta)]
\]
\[
= -2 \pi \int_0^\rho d\rho \int_{-1}^1 d\tau \cos(k \rho \tau)
\]
\[
= -4 \pi k^{-1} \int_0^\rho d\rho \sin(k \rho)
\]
\[
= 4 \pi k^{-2} \left[ \cos(k \rho) - 1 \right].
\] (A11)

Inserting Eqs. (A7)–(A11) into Eq. (A6) gives
\[
\hat{T}(\mathbf{k}) = -\varepsilon_0^{-1} k^{-2} \mathbf{k} \otimes \mathbf{k} \quad \text{for} \quad k \neq 0.
\] (A12)

Taken together, Eqs. (A2) and (A12) prove Eq. (6). The proof of Eq. (14) is a by-product of the above derivation. Here, one has to evaluate the Coulombic potential generated by a periodic array of point dipoles \(\mathbf{\mu}\), which is given by \(\mathbf{t}(\mathbf{r}) \cdot \mathbf{\mu}\)
\[
\mathbf{t}(\mathbf{r}) = \sum_{\mathbf{L} \in \mathbb{Z}} \frac{\mathbf{r} + \mathbf{L} \mathbf{n}}{||\mathbf{r} + \mathbf{L} \mathbf{n}||^3}.
\] (A13)
The corresponding Fourier coefficients are [by analogy with Eq. (A1)],
\[
\hat{\mathbf{t}}(\mathbf{k}) = \int \int_{\mathbb{R}^3} d^3 r \ e^{-i \mathbf{k} \cdot \mathbf{r}} \frac{\mathbf{r}}{r^2}
\]
\[
= -\int \int_{\mathbb{R}^3} d^3 r \ e^{-i \mathbf{k} \cdot \mathbf{r}} \nabla r^{-1}.
\] (A14)
For \(k \neq 0\) the integrand can be rewritten
\[
e^{-i \mathbf{k} \cdot \mathbf{r}} r^{-1} = \nabla (e^{-i \mathbf{k} \cdot \mathbf{r} r^{-1}}) + i \mathbf{k} e^{-i \mathbf{k} \cdot \mathbf{r} r^{-1}}.
\] (A15)
Inserting this expression into Eq. (A14), and using Eqs. (A10) and (A11), one finds
\[
\hat{\mathbf{t}}(\mathbf{k}) = -\lim_{\rho \to \infty} \left[ \mathbf{b}(\mathbf{k}; \rho) - i \mathbf{k} \mathbf{c}(\mathbf{k}; \rho) \right]
\]
\[
= -4 \pi i k^{-2} \mathbf{k}, \quad \text{for} \quad k \neq 0,
\] (A16)
which can be used to derive Eq. (14).

**APPENDIX B: PROOF OF EQ. (12)**

Here, we prove the last equality in Eq. (12). For this purpose, it is sufficient to show that the tensor \(\mathbf{Y}\) defined as
\[
\mathbf{Y} = [1 - \varepsilon_0 (\varepsilon_s - 1)] \hat{T}(\mathbf{k})$
\]
\[
\times \left[ 1 + \varepsilon_0 \frac{\varepsilon_s - 1}{\varepsilon_s} \hat{T}(\mathbf{k})(1 - \delta_k) - \frac{\varepsilon_s - 1}{\varepsilon_s + \frac{1}{2} \delta_k} \right]
\] (B1)
where \(\hat{T}(\mathbf{k})\) is given by Eq. (6), is the unit tensor. For \(k = 0\), Eq. (B1) becomes
\[
\mathbf{Y} = \left[ 1 + \frac{1}{\delta_s} (\varepsilon_s - 1) \right] \left[ 1 - \frac{\varepsilon_s - 1}{\varepsilon_s + \frac{1}{2}} \right] = 1
\] (B2)
which is the expected result. To handle the case \(k \neq 0\), we introduce the tensor \(\mathbf{K} = k^{-2} (\mathbf{k} \otimes \mathbf{k})(1 - \delta_k)\), which satisfies \(\mathbf{K}^2 = \mathbf{K}\). Using this property, Eq. (B1) for \(k \neq 0\) can be rewritten as
\[
\mathbf{Y} = \left[ 1 + (\varepsilon_s - 1) \mathbf{K} \right] \left[ 1 - \frac{\varepsilon_s - 1}{\varepsilon_s} \mathbf{K} \right] = 1
\] (B3)
which is also the expected result. Note that the last equality in Eq. (12) is only proved here for the specific case of a Coulombic dipole–dipole interaction tensor [Eq. (6)], and need not be valid for other (modified) tensors.