Published in final edited form as:
J Comput Phys. 2011 January 20; 230(2): 435–457. doi:10.1016/j.jcp.2010.09.031.

Multiscale molecular dynamics using the matched interface and boundary method

Weihua Geng[1,*] and **G.W. Wei**[1,2,†]

[1] Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA

[2] Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA

Abstract

The Poisson-Boltzmann (PB) equation is an established multiscale model for electrostatic analysis of biomolecules and other dielectric systems. PB based molecular dynamics (MD) approach has a potential to tackle large biological systems. Obstacles that hinder the current development of PB based MD methods are concerns in accuracy, stability, efficiency and reliability. The presence of complex solvent-solute interface, geometric singularities and charge singularities leads to challenges in the numerical solution of the PB equation and electrostatic force evaluation in PB based MD methods. Recently, the matched interface and boundary (MIB) method has been utilized to develop the first second order accurate PB solver that is numerically stable in dealing with discontinuous dielectric coefficients, complex geometric singularities and singular source charges. The present work develops the PB based MD approach using the MIB method. New formulation of electrostatic forces is derived to allow the use of sharp molecular surfaces. Accurate reaction field forces are obtained by directly differentiating the electrostatic potential. Dielectric boundary forces are evaluated at the solvent-solute interface using an accurate Cartesian-grid surface integration method. The electrostatic forces located at reentrant surfaces are appropriately assigned to related atoms. Extensive numerical tests are carried out to validate the accuracy and stability of the present electrostatic force calculation. The new PB based MD method is implemented in conjunction with the AMBER package. MIB based MD simulations of biomolecules are demonstrated via a few example systems.

Keywords

Implicit solvent model; Poisson-Boltzmann equation; Molecular dynamics; Interface method; Matched interface and boundary; Biomolecules

1 Introduction

Since most biological processes, such as gene regulation, signal transduction, and protein-protein interaction occur in aqueous environment, the solvent-solute interactions are important to biological processes. The solvation process is typically described by explicit

[†]Corresponding author. Tel: (517)3534689, Fax: (517)4321562, wei@math.msu.edu.
[*]Present address: Department of Mathematics, University of Michigan, 2074 East Hall, 530 Church Street, Ann Arbor, MI 48109-1043

models and implicit models. Explicit atomistic description of biomolecules and their aqueous environment, including solvent, co-solutes, and mobile ions, is prohibitively expensive for large biomolecular systems, although a variety of advanced computational methods, such as Ewald summations, Euler summations, periodic images and reaction field theory, have been developed for molecular dynamics in the past few decades. In fact, our central interest in most biomolecular studies is the behavior of biomolecules and their interactions, instead of water molecules. As such, implicit solvent models [5,21] that treat the solvent as an electrostatic continuum while admit an atomistic description for biomolecules, are attractive multiscale approaches for the quantitative understanding of large scale biological systems [16,23,32,55,58,62,65,71]. These approaches have become increasingly popular in the biophysical and biochemical community since the earlier work by Warwicker and Watson in the early 1980s [65], and Honig in the 1990s [32]. The most commonly used implicit solvent models include the Poisson-Boltzmann equation (PBE) [5], generalized Born (GB) [22] and polarizable continuum approaches. The most attractive feature of these methods is their combination of atomic details of the biomolecule and a mean field approximation of the solvent. The PBE describes the electrostatic potential under a given charge assignment of the biomolecule and a Boltzmann distribution of the ionic concentration in the solvent. The coulomb gauge for electrostatic potential, the Gauss's divergence theorem and law for the electric field are utilized to derive the PBE. The GB theory is simple and is usually regarded as an approximation to the PBE.

The PBE has a wide range of applications, including the estimation of electrostatic solvation free energies [24,36,59], pK_α values [26,49,64], and the electrostatic analysis of interaction surfaces. There has been a great deal of effort in developing PBE based electrostatic force calculations for molecular dynamics (MD) [28,42,43,45,50,58] in the past two decades. The basic theoretical foundation for such calculations was established by Gilson et al. [28] based on the free energy functional derived by Sharp and Honig [58].

Most implicit solvent models, such as the Poisson-Boltzmann (PB) and GB models, utilize a solvent-molecule interface to separate the continuum domain from that of the biomolecule. Various biomolecular surface models, including the van der Waals surface, the solvent accessible surface [37], the molecular surface (MS) [53], minimal molecular surface [6] and more general geometric flow based biomolecular surfaces [7] are often employed for the application of the PB method. Recently, smoothed molecular surfaces have been proposed for implicit solvent models [14,66,27,76]. Variational approaches are utilized to determine the profile of smoothed surfaces [14,66]. It was argued that care is required when use atomic centered dielectric functions in implicit solvent models for solvation free energy calculations [60,63]. However, as shown by Swanson et al. [61] and in our differential geometry based models [14,66], a simple reparameterization of van der Waals radii can lead to accurate solvation calculations from smoothed surface definitions. Therefore, at a coarse level of approximation, sharp surfaces and smoothed surfaces produce similar results for many PB applications. Thus, the selection of a surface is often dictated by the numerical convenience.

Nevertheless, the numerical instability of the sharp interface becomes a severe problem in the calculation of solvation forces. In molecular dynamics simulations, solvation force has to be evaluated for tens of millions of times. As such, numerical stability is particularly important. To avoid the electrostatic force calculation involving the sharp solvent-solute interfaces, Im et al. presented a smoothed dielectric boundary approach [33]. Using a smooth surface approach, Lu and Luo demonstrated that PBE based methods can be successfully and consistently applied to stable and efficient dynamics simulation of unconstrained biomolecules [43]. Remarkably, Prabhu et al. reported a smooth-permittivity finite difference PB method that is up to eight times faster than the explicit water simulations [50]. However, so far, no force calculation and MD simulation have been

carried out based on any sharp molecular surface definition, although sharp molecular surfaces are widely used in all other PB applications.

The instability of the solvation force evaluation is inherently associated with the instability in the numerical solution of the PBE. The use of sharp molecular surface has certainly contributed to the instability. Many numerical methods have been used to solve the PBE, including finite element methods [3,4,16,30,67], and boundary integral methods [8,9,10,34,39,62,71,44]. The most commonly used PB solvers in the field are usually based on finite difference methods [23,32,42,43,45,58,65,33]. Many popular software packages, such as DelPhi [35,54], UHBD [18], MEAD [54], APBS [30,31,2], AMBER [13] and CHARMM [12,46], adopt finite difference schemes, which offer the best combination of speed, accuracy, and efficiency [5]. Although van der Waals surfaces, solvent accessible surfaces and molecular surfaces were often used in popular software packages, most traditional PB solvers neglect interface flux continuity conditions at the solvent-solute interfaces. As a result, these methods may suffer from instability and slow convergence when directly admit molecular surfaces or van der Waals surfaces. Therefore, stability enhancement and convergence acceleration are pressing technical issues in the development of PB methods.

Mathematically, the aforementioned problem is originated from elliptic equations with discontinuous coefficients and singular sources. This is a very challenging problem in applied mathematics and scientific computing. There has been much research interest in this class of problems since Peskin's pioneer work on immersed boundary method (IBM) [51]. A number of elegant mathematical methods have been introduced in the past two decades, including finite element formulations [1,11,40], the ghost fluid method (GFM) [20,41], coupling interface method [15], and integral equation approach [47]. A remarkable contribution was due to LeVeque and Li [38], who proposed the first second order accurate immersed interface method (IIM). Their method has been applied to the PBE in two spatial dimensions [52], among a variety of other interesting applications. These methods have not been applied to the PBE in biomolecular context. An important problem in computational mathematics is that many existing interface schemes may not work in biomolecular context, due to the complexity in three-dimensional (3D) geometry and geometric singularities.

Recently, we have proposed a high order method, the matched interface and boundary (MIB) method, for elliptic interface problems [72,73,74]. We have demonstrated fourth and sixth order MIB schemes for curved interfaces in 2D [73,74] and 3D [69] applications. More recently, we have developed three generations of MIB based PB solvers, MIBPB-I, MIBPB-II, and MIBPB-III. The MIBPB-I presents a rigorous interface treatment for arbitrarily complex biomolecular interfaces [75] by carefully implementing the flux continuity conditions. However, it cannot maintain the targeted second order convergence because the presence of geometric singularities, such as cusps and self-intersecting surfaces in protein molecular surfaces [17,19,29,56]. To overcome this difficulty, we have developed an MIB method that is able to solve elliptic equations with geometric singularities [68,69]. This MIB scheme based PB solver, called MIBPB-II, is able to deliver the second order accuracy in solving the PBE with molecular surface singularities [70]. The state of the art MIB method is described in our recent paper [69].

Another problem in the development of advanced PB solvers is point charges in proteins, which are represented by Dirac delta functions. As distributions, delta functions reduce the regularity of the solution of the PBE. In most existing PB solvers, Dirac delta functions in the PBE, or singular charges in a molecule, are redistributed to their neighboring grid points. This approach is compatible with the neglect of interface flux continuity conditions in commonly used PB solvers. However, in the MIBPB-II, accuracy reduction occurs when a

grid point has both redistributed charge contribution and interface condition contribution. This typically happens when the mesh size approaches half of the van der Waals radius, i.e., about 0.6Å, for smallest atoms in a protein. To overcome this difficulty, a regularization procedure that separates the solution of the PBE into a regular part and an irregular part was employed in our MIBPB-III [25]. The irregular part consists of the Green's function, which is the solution to the Poisson equation with the singular charges. The MIBPB-III was found to be a few orders of magnitude more accurate and up to three times faster than commonly used PB solvers in achieving the same accuracy [25].

The work described in this paper is a continuation of our endeavor in developing mathematical methods for problems in molecular biology. The objective of the present work is to develop stable and accurate algorithms for electrostatic force calculations based on the sharp molecular surfaces. The MIBPB technique is utilized to solve the PBE with molecular surfaces. The present work offers a number of new features. First, the present force calculation is based on the molecular surface [53] or the minimal molecular surface [6], without resorting to any surface preprocessing, such as the geometric smoothening and/or the dielectric smoothening. Additionally, to directly evaluate solvation forces with sharp interfaces, force density expressions, particularly the dielectric boundary force density, are derived. The theoretical approach used in our present derivation is similar to that used in our differential geometry based multiscale models [66]. Yet, the resulting force density involves the evaluation of limiting values of the first order derivatives of the electrostatic potential from both inside and outside the interface. Moreover, the molecular surface interface contains reentrant surfaces which are detached from individual atomic spheres. Therefore, forces from reentrant surfaces need to be distributed to nearby atoms, which touch the probe when it forms the reentrant surface. Furthermore, accurate surface integrations are required for the evaluation of dielectric boundary forces and ionic forces. These evaluations are to be carried out along the complex interface and in the Cartesian grid. Finally, a PB solver that is able to handle sharp interface without numerical instability is needed. This is done with our MIBPB-II method.

The rest of this paper is organized as the follows. Theory and algorithm are presented in Section 2. The derivation of electrostatic force density functions is based on a free energy functional. Derivation of the density functions is discussed with respect to sharp interfaces and molecular surfaces. A number of numerical algorithms, including the evaluation of the force density as limiting values at the interface, accurate surface integrations in Cartesian grids, and the distribution of dielectric boundary forces to the individual atoms, are developed. Validation and numerical experiments are described in Section 3. The method of accurate integration in Cartesian grid is tested with the area of molecular surfaces. The proposed force algorithms are tested over many polyatomic systems. Section 4 is devoted to the molecular dynamics applications. The present electrostatic forces are implemented in conjugation with the AMBER force fields [13], particularly, in place of the AMBER PB module [43,45] for molecular dynamics simulations. Three small biomolecules are used in our demonstration. This paper ends with concluding remarks summarizing the results.

2 Theory and algorithm

This section describes the basic theories and algorithms that are required for the construction of the proposed MIBPB based MD method. Electrostatic free energy functional and electrostatic force were presented by Sharp and Honig [58], and Gilson et al. [28], respectively. In the present work, we present new derivations that are appropriate for sharp solvent-solute interfaces. Section 2.1 is devoted to the free energy functional for electrostatic problems. In Section 2.2, we present a new derivation of electrostatic forces. Force expressions for surface integrations are given in Section 2.3. These new force expressions

involve left and right gradients of the electrostatic potential on the molecular surface, which are to be computed by using the MIBPB method. To establish notation, Section 2.4 gives a brief review of the MIBPB method. Finally, Section 2.5 provides details for calculating dielectric boundary forces. Such calculations involve three techniques: gradient evaluation, surface integral on Cartesian grid, and the distribution of force elements to related atoms.

2.1 Free energy functional

Consider an open domain $\Omega \in \mathbb{R}^3$. Let Γ be the interfaces which divides Ω into two disjoint open subdomains, $\Omega = \Omega^- \cup \Omega^+ \cup \Gamma$. Here Ω^- is the biomolecular domain, and Ω^+ is the solvent domain. The starting point for the implicit solvent theory is a given set of atomic coordinates for a molecule in $\mathbb{R}^3$. The charge density of the molecule $\rho_m(\mathbf{r})$ can therefore be obtained by assigning charge to each atom according to its type and environment

$$\rho_m = 4\pi \sum_{i=1}^{N_m} Q_i \delta(\mathbf{r} - \mathbf{r}_i), \tag{1}$$

where Q_i is the partial charges of ith atom, and N_m is the number of charged atoms. The positions of the charge $\mathbf{r}_i$ is determined from experiments, such as X-ray crystallography, nuclear magnetic resonance and computations. We introduce the biomolecular domain characteristic function

$$\lambda^-(\mathbf{r}) = \begin{cases} 1 & \mathbf{r} \in \Omega^- \cup \Gamma; \\ 0 & \mathbf{r} \in \Omega^+. \end{cases} \tag{2}$$

Obviously, $\lambda^+ = (1 - \lambda^-)$ characterizes the solvent domain Ω^+. The permittivity $\varepsilon(\mathbf{r})$

$$\varepsilon(\mathbf{r}) = \lambda^- \varepsilon^- + \lambda^+ \varepsilon^+ \tag{3}$$

is a function of position $\mathbf{r} \in \mathbb{R}^3$, and its value in the molecular domain (Ω^-) differs from that in the solvent domain (Ω^+). Note that Eqs. (2) and (3) present a simplified treatment of the ion domain. More appropriate treatment should include an additional ion exclusion zone outside the biomolecular domain [25].

The charge density of the solvent $\rho_s(\varphi(\mathbf{r}))$ is a function of the electrostatic potential φ. In case of a multi-species system, a commonly used ionic density is the Boltzmann distribution and is given by

$$\rho_s = \sum_{j=1}^{N} q_j c_j e^{-q_j \varphi / kT}, \tag{4}$$

where N is the number of ionic species, c_j's are the bulk concentration of each ionic species, and q_j's are charges of each ionic species. The free energy of the electrostatic system can be expressed as an integral

$$G^{\text{elec}} = \int_{\Omega} g^{\text{elec}}(x, y, z, \varphi, \varphi_x, \varphi_y, \varphi_z) d\Omega, \tag{5}$$

where g^{elec} is the free energy functional or energy density

$$g^{\text{elec}} = \lambda^- \rho_m \varphi - \lambda^- \frac{\varepsilon^-}{2} |\nabla \varphi|^2 - \lambda^+ \frac{\varepsilon^+}{2} |\nabla \varphi|^2 + \lambda^+ \bar{\rho}_s. \tag{6}$$

Here $\bar{\rho}_s(\varphi) = \int \rho_s(\varphi) d\varphi$ and $\bar{\rho}_s(0) = 0$. If $\rho_s(f)$ is given by Eq. (4), then we have

$$\bar{\rho}_s = -kT \sum_{j=1}^{N} [c_j (e^{-q_j \varphi / kT} - 1)]. \tag{7}$$

To minimize the free energy, the energy functional $g^{\text{elec}}(x, y, z, \varphi, \varphi_x, \varphi_y, \varphi_z)$ must satisfy the Euler-Lagrange equation

$$\frac{\partial g^{\text{elec}}}{\partial \varphi} - \left[\frac{\partial}{\partial x} \frac{\partial g^{\text{elec}}}{\partial \varphi_x} + \frac{\partial}{\partial y} \frac{\partial g^{\text{elec}}}{\partial \varphi_y} + \frac{\partial}{\partial z} \frac{\partial g^{\text{elec}}}{\partial \varphi_z} \right] = 0, \tag{8}$$

which gives rise to the PBE,

$$-\nabla \cdot \varepsilon(\mathbf{r}) \nabla \varphi(\mathbf{r}) = \lambda^- \rho_m + \lambda^+ \rho_s. \tag{9}$$

Obviously, there are flexibilities in the selection of ρ_s. In particular, the present choice does not account for the effect of ionic correlation and ion-macromolecule interaction near the interface.

For a sharp interface Γ, the permittivity values $\varepsilon(\mathbf{r})$ at molecular domain Ω^- and the solvent domain Ω^+ are different, as shown in Eq. (3). In most calculations, ε^- and ε^+ are set to 1 and 80, respectively. This jump in permittivity gives rise to discontinuous coefficients in the PBE (9). To ensure the uniqueness of the solution, the PBE is to be solved with the following continuity conditions at the interface

$$[\varphi] = \varphi^+(\mathbf{r}) - \varphi^-(\mathbf{r}) = 0, \tag{10}$$

$$[\varepsilon \varphi_{\mathbf{n}}] = \varepsilon^+(\mathbf{r}) \nabla \varphi^+(\mathbf{r}) \cdot \mathbf{n} - \varepsilon^-(\mathbf{r}) \nabla \varphi^-(\mathbf{r}) \cdot \mathbf{n} = 0, \tag{11}$$

where $\mathbf{n} = (n_x, n_y, n_z)$ is the outer normal direction and $\varphi^{\pm}$ are the limiting values at the interface Γ. Here $[\varphi] = 0$ and $[\varepsilon \varphi_{\mathbf{n}}] = 0$ for the PBE therefore $[\varphi_{\mathbf{n}}]$ are non-zero. The interface normal can be computed from given interfaces, such as the molecular surface of a given protein.

2.2 Electrostatic forces

The electrostatic force at a reference point, say the origin, denoted as $\mathbf{F}^{\text{elec}}$, can be expressed as the volume integral of the gradient of the free energy density g^{elec}

$$\mathbf{F}^{\text{elec}} = -\int_{\Omega} \nabla g^{\text{elec}} d\Omega. \tag{12}$$

In this treatment, we assume that the system is at equilibrium so that the temperature T, the ion concentration c_i and their charge q_i are position independent quantities. From Eqs. (6) and (7), we have

$$\mathbf{F}^{\text{elec}} = -\int_{\Omega} \left(\lambda^- \varphi \nabla \rho_m + \varphi \rho_m \nabla \lambda^- - \frac{1}{2} |\nabla \varphi|^2 \nabla \varepsilon - kT \sum_{j=1}^{N} [c_j (e^{-q_j \varphi / kT} - 1)] \nabla \lambda^+ + \lambda^- \rho_m \nabla \varphi - \varepsilon \nabla \varphi \cdot \nabla \nabla \varphi + \lambda^+ \sum_{j=1}^{N} [q_j c_j e^{-q_j \varphi / kT} \nabla \varphi] \right) d\Omega. \tag{13}$$

The first term on the right hand side of Eq. (13) can be integrated by parts

$$\int_{\Omega} \lambda^- \varphi \nabla \rho_m d\Omega = -\int_{\Omega} \rho_m \nabla (\lambda^- \varphi) d\Omega, \tag{14}$$

where the product $\varphi \rho_m$ vanishes at infinity. Using the product rule on the sixth term of Eq. (13), we have

$$\int_{\Omega} (\varepsilon \nabla \varphi) \cdot (\nabla \nabla \varphi) d\Omega = \int_{\Omega} \nabla \cdot (\varepsilon \nabla \varphi \nabla \varphi) d\Omega - \int_{\Omega} (\nabla \varphi) \nabla \cdot (\varepsilon \nabla \varphi) d\Omega. \tag{15}$$

The first term on the right vanishes as a surface integral at infinity. The resulting force expression becomes

$$\mathbf{F}^{\text{elec}} = -\int_{\Omega} \left(-\rho_m \nabla (\lambda^- \varphi) + \varphi \rho_m \nabla \lambda^- - \frac{1}{2} \mathbf{E}^2 \nabla \varepsilon - kT \sum_{j=1}^{N} c_j (e^{-q_j \varphi / kT} - 1) \nabla \lambda^+ + \left[\lambda^- \rho_m + \nabla \cdot (\varepsilon \nabla \varphi) + \lambda^+ \sum_{j=1}^{N} q_j c_j e^{-q_j \varphi / kT} \right] (\nabla \varphi) \right) d\Omega, \tag{16}$$

where $\mathbf{E} = -\nabla \varphi$ is the electrostatic field vector. Recognizing that the terms in the square bracket vanish because of the PBE (9), one arrives at the force expression

$$\mathbf{F}^{\text{elec}} = -\int_\Omega \left(-\rho_m \nabla(\lambda^- \varphi) + \varphi \rho_m \nabla \lambda^- - \frac{1}{2}\mathbf{E}^2 \nabla \varepsilon - kT \sum_{j=1}^{N} c_j (e^{-q_j \varphi / kT} - 1) \nabla \lambda^+ \right) d\Omega = \int_\Omega \mathbf{f}^{\text{elec}} d\Omega, \tag{17}$$

where the force density is

$$\mathbf{f}^{\text{elec}} = \rho_m \nabla(\lambda^- \varphi) - \varphi \rho_m \nabla \lambda^- + \frac{1}{2}\mathbf{E}^2 \nabla \varepsilon + kT \sum_{j=1}^{N} c_j (e^{-q_j \varphi / kT} - 1) \nabla \lambda^+. \tag{18}$$

This expression is to be further simplified for practical calculations.

Since $\nabla \lambda^-$ gives rise to a Dirac delta function at the solvent-solute interface, the second term $\varphi \rho_m \nabla \lambda^-$ vanishes because ρ_m consists of delta functions away from the interface. Similarly, the first term can be simplified as $\lambda^- \rho_m \nabla \varphi$. Therefore, the total force can be broken into three contributions $\mathbf{F}^{\text{elec}} = \mathbf{F}^{\text{RF}} + \mathbf{F}^{\text{DB}} + \mathbf{F}^{\text{IB}}$. The first term is the reaction field force at the reference point and is often evaluated at the charge locations

$$\begin{aligned} \mathbf{F}^{\text{RF}} &= \int_\Omega \lambda^- \rho_m \nabla \varphi d\Omega \\ &= \int_{\Omega^-} \rho_m \nabla \varphi d\Omega \\ &= 4\pi \sum_{i=1}^{N_m} \int_{\Omega^-} Q_i \delta(\mathbf{r} - \mathbf{r}_i) \nabla \varphi d\Omega \\ &= 4\pi \sum_{i=1}^{N_m} Q_i (\nabla \varphi)|_{\mathbf{r}=\mathbf{r}_i}. \end{aligned} \tag{19}$$

In practical computations, $\mathbf{F}^{\text{RF}}$ corresponding to the solvation is evaluated as the difference of the solution gradients in solvent and in vacuum

$$\begin{aligned} \mathbf{F}^{\text{RF}} &= 4\pi \sum_{i=1}^{N_m} Q_i [(\nabla \varphi_s)|_{\mathbf{r}=\mathbf{r}_i} - (\nabla \varphi_v)|_{\mathbf{r}=\mathbf{r}_i}] \\ &= \sum_{i=1}^{N_m} \mathbf{F}_i^{\text{RF}}, \end{aligned} \tag{20}$$

where $\mathbf{F}_i^{\text{RF}}$ is the reaction field force at ith atom, φ_s is the potential for solvated molecule and φ_v is the that of molecule in vacuum. Note that it is $\mathbf{F}_i^{\text{RF}}$, instead of $\mathbf{F}^{\text{RF}}$, that is evaluated and used in molecular dynamics. The second term in Eq. (17) is the dielectric boundary force at the reference point

$$\mathbf{F}^{\text{DB}} = \int_\Omega \frac{1}{2}\mathbf{E}^2 \nabla \varepsilon d\Omega. \tag{21}$$

The last term in Eq. (17) is the ionic boundary force at the reference point

$$\mathbf{F}^{\mathrm{IB}}=\int_{\Omega}kT\sum_{i}^{N}c_i(e^{-q_i\varphi/kT}-1)\nabla\lambda^{+}d\Omega. \tag{22}$$

Both the dielectric boundary force and the ionic boundary force are to be evaluated on the surface of the biomolecule.

2.3 Force expressions involving sharp interfaces

For the dielectric boundary force, the force expression involves the gradients of the characteristic functions

$$\nabla\varepsilon(\mathbf{r})=\varepsilon^{-}\nabla\lambda^{-}+\varepsilon^{+}\nabla\lambda^{+}. \tag{23}$$

According to Eq. (2), these gradients vanish away from the interface Γ. Let **R**(Γ) be a general position vector on the interface Γ, and

$$\mathbf{R}^{\pm}(\Gamma)=\lim_{\varepsilon\to 0}\mathbf{R}(\Gamma)\pm\varepsilon\hat{\mathbf{n}}, \tag{24}$$

where $\hat{\mathbf{n}}$ is the unit normal vector at **R**(Γ). Note that $\mathbf{R}^{\pm}(\Gamma)$ vary only along the positive and the negative $\hat{\mathbf{n}}$ directions. We therefore have

$$\nabla\lambda^{\pm}(\mathbf{r})=\begin{cases}\pm\hat{\mathbf{n}}\delta(r-\mathbf{R}^{\pm}(\Gamma)) & \mathbf{r}\ \text{near}\ \Gamma;\\ \mathbf{0} & \mathbf{r}\ \text{away from}\ \Gamma,\end{cases} \tag{25}$$

where *r* is the position variable in the unit normal direction. The dielectric boundary force at the reference point is

$$\mathbf{F}^{\mathrm{DB}}=\int_{\Gamma}\int\tfrac{1}{2}\mathbf{E}^{2}[\varepsilon^{+}\delta(r-\mathbf{R}^{+}(\Gamma))-\varepsilon^{-}\delta(r-\mathbf{R}^{-}(\Gamma))]\hat{\mathbf{n}}drdS. \tag{26}$$

By carrying out the integration over *r*, we have

$$\mathbf{F}^{\mathrm{DB}}=\int_{\Gamma}\mathbf{f}^{\mathrm{DB}}dS \tag{27}$$

$$=\int_{\Gamma}\frac{1}{2}(\varepsilon^{+}|\mathbf{E}^{+}|^{2}-\varepsilon^{-}|\mathbf{E}^{-}|^{2})d\mathbf{S}, \tag{28}$$

where $d\mathbf{S}=\hat{\mathbf{n}}dS$ is pointed at the surface normal direction **n** and $\mathbf{f}^{\mathrm{DB}}=\frac{1}{2}(\varepsilon^{+}|\mathbf{E}^{+}|^{2}-\varepsilon^{-}|\mathbf{E}^{-}|^{2})\hat{\mathbf{n}}$ is the dielectric boundary force density. Here, $\mathbf{E}^{\pm}=-\nabla\varphi^{\pm}$ are to be evaluated at the interface Γ. It is noted that the full-scale integration of Eq. (28) will not be pursued. It is the local forces at all intersecting points of the interface and meshlines that are needed. Such forces

are evaluated by $\frac{1}{2}(\varepsilon^+|\mathbf{E}^+|^2 - \varepsilon^-|\mathbf{E}^-|^2)\widehat{\mathbf{n}}\Delta S$, where ΔS is the area of the surface element in a given mesh.

For the ionic boundary force $\mathbf{F}^{\mathrm{IB}}$, since λ^+ has its non-zero value only in the solvent, $\nabla\lambda^+$ contributes to a delta function evaluated at points on the interface where the mesh lines and the interface intersect. Consequently, the ionic boundary force is

$$\mathbf{F}^{\mathrm{IB}} = \int_{\Gamma} \mathbf{f}^{\mathrm{IB}} dS \tag{29}$$

$$= \int_{\Gamma} kT \sum_{j=1}^{N} c_j (e^{-q_j\varphi^+/kT} - 1) d\mathbf{S}, \tag{30}$$

where φ^+ is to be evaluated at the interface Γ and $\mathbf{f}^{\mathrm{DB}} = kT\sum_{j=1}^{N} c_j(e^{-q_j\varphi^+/kT} - 1)\widehat{\mathbf{n}}$ is the ionic boundary force density. At these intersecting points, the values of φ^+ can be obtained by the interpolation of two grid points outside the interface and a fictitious point inside the interface. When singular geometry occurs, interpolation using one grid point outside the interface and two fictitious points inside can provide an alternative.

Here we mention that when a more appropriate ion exclusion zone is included, the notation will become more complicated, but the derivation of the ionic boundary force is the same. The calculation of the ionic boundary force is very similar to that of the dielectric boundary force technically but it has less impact on the total electrostatic solvation force. Therefore, we just mention this aspect briefly here without providing further consideration and numerical results.

In this paper, we carry out dielectric boundary force evaluations by using technique based on sharp molecular surfaces, which is developed in the next subsection. The molecular surface is generated by using the MSMS [56]. A technique developed in Ref. [75] is used to convert the triangle surface representation from the MSMS into the required Cartesian representation.

2.4 The MIB solution of the Poisson Boltzmann equation

Rigorous interface technique, the MIB method, has been described in a series work for the solution of Eq. (31) [25,70,75]. To establish notation, we present a brief description of our MIB based PB solvers.

Assume that an interface Γ divides a domain Ω into two disjoint parts, a molecular subdomain Ω^- and a solvent subdomain Ω^+. The outer boundary of Ω^+ is chosen to be regular. Without loss of generality, let us consider the linearized Poisson Boltzmann equation (PBE) of the form

$$-\nabla\cdot(\varepsilon(\mathbf{r})\nabla\varphi(\mathbf{r})) + \kappa^2(\mathbf{r})\varphi(\mathbf{r}) = 4\pi\lambda^-\sum_{i=1}^{N_m} Q_i\delta(\mathbf{r}-\mathbf{r}_i), \tag{31}$$

where the ionic strength $\kappa(\mathbf{r})$ is defined as $\kappa^2=\frac{\lambda^+(\mathbf{r})}{\varepsilon kT}\sum_{j=1}^{N}c_j q_j^2$. The continuity conditions of the solution and its flux across the interface are given by Eqs. (10) and (11), respectively. In the MIB method, these conditions are to be implemented rigorously in order to obtain the highly accurate and fast convergent solution to the PBE. Dirichlet far-field boundary condition $\varphi(\infty) = 0$ is used for Eq. (31).

The essence of MIBPB solvers is the treatment of interface continuity conditions Eqs. (10) and (11) in the Cartesian grid. The standard finite difference schemes cannot maintain the designed order of convergence due to the discontinuity in the flux. To restore the convergence, we compute the derivatives in the PBE near the interface without directly using the function values across the interface. Instead, we use only function values on the same side of the interface and their smooth extensions across the interface, which are often called fictitious values, see Fig. 1. A fictitious value is needed when a finite difference scheme reaches across the interface. To determine fictitious values, we make use of continuity conditions at the interface. This in turn enforces the continuity conditions.

The MIB method takes the dimension splitting approach which reduces a multidimensional interface problem into 1D ones. Therefore, unlike the IIM, the MIB method normally does not use multidimensional interpolation or multidimensional Taylor expansion. Consequently, the MIB method simplifies the local topological relation near an interface, which is crucial for 3D problems with complex interface geometries.

Consider a case where the interface Γ intersects the grid in the x-direction at a point (i_o, j, k), which has two nearest neighbor grid points (i, j, k) and $(i + 1, j, k)$. Fictitious values $f^+(i, j, k)$ and $f^-(i + 1, j, k)$ are required in a second order finite difference scheme in the x-direction near the intersecting point. In general, at every intersecting point of the interface and mesh lines, there are two original interface conditions and two additional first order interface conditions. These four conditions can in principle determine four fictitious values. However, these four interface conditions involve six first order derivatives $\varphi_x^+, \varphi_x^-, \varphi_y^+, \varphi_y^-, \varphi_z^+$, and φ_z^-. The evaluation of these derivatives can be very difficult for a complex geometry. In the MIB method, we simultaneously determine as few fictitious values around an intersecting point as possible so that we have the maximal flexibility in avoiding the determination of many first order derivatives, which are often difficult to evaluate due to the geometric constraint. Nevertheless, we have to determine at least two fictitious values so that both of the original two interface conditions can be implemented directly or indirectly. Consequently, we determine only two fictitious values around an intersecting point and thus, use two interface conditions to eliminate two first order derivatives. The remaining four derivatives are to be approximated using appropriate grid function values near the intersecting point.

In order to derive two additional conditions, we define a local coordinate (ξ,η, ζ) such that ξ is along the normal direction, and η is in the $x – y$ plane. Two coordinates are connected by the transformation

$$\begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} = \begin{bmatrix} \sin\psi\cos\theta & \sin\psi\sin\theta & \cos\psi \\ -\sin\theta & \cos\theta & 0 \\ -\cos\psi\cos\theta & -\cos\psi\sin\theta & \sin\psi \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{p} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \tag{32}$$

where θ and ψ are the azimuth and zenith angles with respect to the normal direction ξ, respectively. Two additional continuity conditions are generated by differentiating Eq. (10) along two tangential directions, η and ζ

$$[\varphi_\eta]=(\varphi_x^+ p_{21}+\varphi_y^+ p_{22}+\varphi_z^+ p_{23}) - (\varphi_x^- p_{21}+\varphi_y^- p_{22}+\varphi_z^- p_{23}) \tag{33}$$

$$[\varphi_\zeta]=(\varphi_x^+ p_{31}+\varphi_y^+ p_{32}+\varphi_z^+ p_{33}) - (\varphi_x^- p_{31}+\varphi_y^- p_{32}+\varphi_z^- p_{33}), \tag{34}$$

where p_{ij} is the ijth component of the transformation matrix **p**. From Eq. (32) we have $[\varphi_\xi, \varphi_\eta, \varphi_\zeta]^T = \mathbf{p} \cdot [\varphi_x, \varphi_y, \varphi_z]^T$, where T denotes the transpose. Three flux continuity conditions, Eqs. (11), (33) and (34) can be cast in the form

$$\begin{bmatrix} [\varepsilon\varphi_\xi] \\ [\varphi_\eta] \\ [\varphi_\zeta] \end{bmatrix} = \mathbf{C} \cdot \begin{bmatrix} \varphi_x^+ \\ \varphi_x^- \\ \varphi_y^+ \\ \varphi_y^- \\ \varphi_z^+ \\ \varphi_z^- \end{bmatrix}, \tag{35}$$

where

$$\mathbf{C}=\begin{bmatrix} \mathbf{C}_1 \\ \mathbf{C}_2 \\ \mathbf{C}_3 \end{bmatrix} = \begin{bmatrix} p_{11}\varepsilon^+ & -p_{11}\varepsilon^- & p_{12}\varepsilon^+ & -p_{12}\varepsilon^- & p_{13}\varepsilon^+ & -p_{13}\varepsilon^- \\ p_{21} & -p_{21} & p_{22} & -p_{22} & p_{23} & -p_{23} \\ p_{31} & -p_{31} & p_{32} & -p_{32} & p_{33} & -p_{33} \end{bmatrix}, \tag{36}$$

where $\mathbf{C}_i$ represents the ith row of matrix **C**. As discussed above, there are six derivatives, $\varphi_x^+, \varphi_x^-, \varphi_y^+, \varphi_y^-, \varphi_z^+$, and φ_z^-, in restricted subdomains. Some of these conditions are very difficult to evaluate for complex biomolecular interface geometries. Therefore, in the MIB method, we eliminate two derivatives as the follows. In Fig. 1(b) we illustrate a case where the φ_y^- is difficult to compute and is to be eliminated. In general, after the elimination of the lth and mth elements of the array $[\varphi_x^+, \varphi_x^-, \varphi_y^+, \varphi_y^-, \varphi_z^+, \varphi_z^-]$, Eq. (35) becomes

$$a[\varepsilon\varphi_\xi]+b[\varphi_\eta]+c[\varphi_\zeta]=(a\mathbf{C}_1+b\mathbf{C}_2+c\mathbf{C}_3) \cdot \begin{bmatrix} \varphi_x^+ \\ \varphi_x^- \\ \varphi_y^+ \\ \varphi_y^- \\ \varphi_z^+ \\ \varphi_z^- \end{bmatrix}, \tag{37}$$

where

$$\begin{aligned} a&=C_{2l}C_{3m} - C_{3l}C_{2m} \\ b&=C_{3l}C_{1m} - C_{1l}C_{3m} \\ c&=C_{1l}C_{2m} - C_{2l}C_{1m}. \end{aligned} \tag{38}$$

Finally, Eqs. (10) and (37) are used to determine two fictitious values near the interface, together with the evaluation of four partial derivatives. This procedure is systematically repeated to determine fictitious values along other mesh lines and at other intersecting points. This dimension splitting approach effectively reduces a 3D interface problem into 1D-like ones locally.

For the situation depicted in Fig. 1(b), after elimination of φ_y^-, we compute φ_y^+. However, on the line passing through (i_o, j, k) in the y-direction, there is no grid point. Therefore, φ_y^+ has to be evaluated by using interpolation schemes with solution values from the neighboring points. Specifically, φ_y^+ at (i_o, j, k) will be interpolated by using values at points $(i_o, j-1, k)$, (i_o, j, k) and $(i_o, j+1, k)$. Because these three points are not normal grid points, we have to further obtain the values at $(i_o, j-1, k)$ by interpolation using grid values at $(i+1, j-1, k)$, $(i+2, j-1, k)$ and $(i+3, j-1, k)$, and the value at $(i_o, j+1, k)$ by using grid values at $(i+1, j+1, k)$, $(i+2, j+1, k)$ and $(i+3, j+1, k)$. For the value at (i_o, j, k), the fictitious value at (i, j, k) and grid values at $(i+1, j, k)$ and $(i+2, j, k)$ will be employed for the interpolation.

There are some simple principles for the selection of two derivatives to be eliminated and for the selection of the interpolation strategy. One is the feasibility of the scheme and the other is matrix optimization. We eliminate those derivatives that are difficult to evaluate. For a given local geometry, we will choose to compute one of two partial derivatives, φ_z^+ and φ_z^-, such that the evaluation in the $x-z$ plane can be easily carried out. We select an appropriate interpolation strategy such that the resulting MIB matrix is optimally symmetric and diagonal for arbitrarily complex solvent-molecule interfaces with geometric singularities. A detailed discussion of the matrix optimization and 3D MIB methods for geometric singularities can be found in Ref. [69]. A rigorous validation of MIB based PB solver, MIBPB-II, for molecular surface singularities can be found in Ref. [70]. For a detailed treatment of charge singularities, the reader is referred to Ref. [25].

2.5 Dielectric boundary force calculation

There are two features in the present work that differ from any of the previous work in electrostatic force calculation. First, our force calculation is based on the rigorous treatment of the PBE with discontinuous solvent-solute interfaces. Therefore, our evaluation of the dielectric boundary force density is to be based on Eq. (28), instead of Eq. (21), for sharp interfaces. Second, due to the use of molecular surfaces, instead of van der Waals surfaces, the distribution of reentrant surface effects in the dielectric boundary force to individual atoms is a new issue. In the present calculation of the dielectric boundary force, three issues are to be resolved. First, one needs to evaluate $\mathbf{E}^{\pm}$ in Eq. (28), which involves partial derivatives at the interface. Second, one needs to numerically carry out the surface integral Eq. (28) along the interface. Finally, one needs to distribute the calculated boundary forces from reentrant surfaces to individual atoms. For the first issue, the calculation of the surface density function is actually at the intersecting points of the interface and the meshlines. Some of these intersecting point values are actually computed in solving PBE, and by modifying part of our MIB schemes. Derivatives of φ with respect to x, y and z-directions on these points in both inside and outside of the interface can be conveniently computed. For the surface integration, we employ the method developed by Smereka [57] with a minor modification. In this method, the numerical surface integral is calculated by volume integral with integrand supported by discrete delta functions defined only on grid points close to the interface. For the issue of force distribution, since our numerical integral is in the form of the summation of the product of the force density and curved small area, we can distribute the force to individual atoms based on the location of the force density. The details of

surface density evaluation, surface integration and force distribution are given in the following three subsections.

2.5.1 Evaluation of dielectric boundary force density on the interface—The dielectric boundary force on the interface as discussed in the previous section is given by Eq. (28). To compute $\mathbf{E}^+$ and $\mathbf{E}^-$, we need to evaluate the gradient of φ on the interface from both inside of the biomolecule denoted as "−", and outside of the biomolecule but in the solvent denoted as "+", i.e., $\varphi_y^-, \varphi_z^+, \varphi_z^-, \varphi_x^+, \varphi_x^-$ and φ_y^+.

Having obtained the solution φ to the PBE in the entire domain and supposing now one wants to evaluate these partial derivatives at (x_o, y_j, z_k) from Fig. 1(b), φ_x^+ can be interpolated by values of φ at $(i-1, j, k)$ and (i, j, k), and the fictitious value of φ at $(i+1, j, k)$, which is the linear combination of values at some grid points. Meanwhile φ_x^- can be interpolated by the values of φ at $(i+1, j, k)$ and (i, j, k), and the fictitious value of at $(i-1, j, k)$. In our MIB scheme, two of $\varphi_y^+, \varphi_y^-, \varphi_z^+$ and φ_z^- can also be interpolated by values at one fictitious point, two grid points and six auxiliary points (actually just regular grid points). Here, we only need one derivative from these two, i.e., φ_y^+. Then, one can use Eq. (35) to compute φ_y^-, φ_z^+ and φ_z^-.

To this end, one rewrites Eq. (36) into the following form by block matrices

$$\mathbf{J}=\begin{bmatrix}[\varepsilon\varphi_\xi]\\ [\varphi_\eta]\\ [\varphi_\zeta]\end{bmatrix}=\mathbf{C_A}\cdot\begin{bmatrix}\varphi_x^+\\ \varphi_x^-\\ \varphi_y^+\end{bmatrix}+\mathbf{C_B}\cdot\begin{bmatrix}\varphi_y^-\\ \varphi_z^+\\ \varphi_z^-\end{bmatrix} \tag{39}$$

where $\mathbf{C_A}$ and $\mathbf{C_B}$ are the first three and last three columns of C, respectively. With this setting, one can obtain the three unknown partial derivatives by

$$\begin{bmatrix}\varphi_y^-\\ \varphi_z^+\\ \varphi_z^-\end{bmatrix}=\mathbf{C_B^{-1}}\cdot\left(\mathbf{J}-\mathbf{C_A}\cdot\begin{bmatrix}\varphi_x^+\\ \varphi_x^-\\ \varphi_y^+\end{bmatrix}\right). \tag{40}$$

Note that under some rare situations, matrix $\mathbf{C_B}$ could be singular, and one needs to first compute four of six partial derivatives of φ by an interpolation with grid point and fictitious values, and then use the interface jump conditions to solve the remaining two. Moreover, if an intersecting point is exactly a grid point, some special treatments are required as described in the MIB schemes [68,69]. These treatments are implemented in our software package. Their description is lengthy and is omitted here.

2.5.2 Numerical surface integration in Cartesian grids—According to Smereka, the surface integral of a density function in Cartesian grids can be approximated by [57]

$$\int_\Gamma f(x,y,z)dS=\int_\Omega f(x,y,z)\delta(d(x,y,z))d\Omega\approx\sum_{i,j,k}f(x_i,y_j,z_k)\tilde{\delta}_{i,j,k}h^3 \tag{41}$$

where (x_i, y_j, z_k) is the coordinate of grid point (i, j, k), $d(x, y, z)$ is the distance of a point (x, y, z) defined in Ω from Γ, and $f(x, y, z)$ is the surface density function defined in Γ. With Φ as the level set function such that $\Phi = 0$ defines the interface, i.e. $\Gamma = \{(x, y, z): \Phi(x, y, z) = 0\}$, the first order discrete delta function at (i, j, k) is given by [57]

$$\tilde{\delta}_{i,j,k} = \tilde{\delta}^{(+x)}_{i,j,k} + \tilde{\delta}^{(-x)}_{i,j,k} + \tilde{\delta}^{(+y)}_{i,j,k} + \tilde{\delta}^{(-y)}_{i,j,k} + \tilde{\delta}^{(+z)}_{i,j,k} + \tilde{\delta}^{(-z)}_{i,j,k}, \tag{42}$$

where

$$\tilde{\delta}^{(+x)}_{i,j,k} = \begin{cases} \frac{|\Phi_{i+1,j,k} D_x^0 \Phi_{i,j,k}|}{h^2 |D_x^+ \Phi_{i,j,k}| \|\nabla_0^\varepsilon \Phi_{i,j,k}|} & \text{if } \Phi_{i,j,k}\Phi_{i+1,j,k} \le 0; \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{\delta}^{(-x)}_{i,j,k} = \begin{cases} \frac{|\Phi_{i-1,j,k} D_x^0 \Phi_{i,j,k}|}{h^2 |D_x^- \Phi_{i,j,k}| \|\nabla_0^\varepsilon \Phi_{i,j,k}|} & \text{if } \Phi_{i,j,k}\Phi_{i-1,j,k} < 0; \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{\delta}^{(+y)}_{i,j,k} = \begin{cases} \frac{|\Phi_{i,j+1,k} D_y^0 \Phi_{i,j,k}|}{h^2 |D_y^+ \Phi_{i,j,k}| \|\nabla_0^\varepsilon \Phi_{i,j,k}|} & \text{if } \Phi_{i,j,k}\Phi_{i,j+1,k} \le 0; \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{\delta}^{(-y)}_{i,j,k} = \begin{cases} \frac{|\Phi_{i,j-1,k} D_y^0 \Phi_{i,j,k}|}{h^2 |D_y^- \Phi_{i,j,k}| \|\nabla_0^\varepsilon \Phi_{i,j,k}|} & \text{if } \Phi_{i,j,k}\Phi_{i,j-1,k} < 0; \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{\delta}^{(+z)}_{i,j,k} = \begin{cases} \frac{|\Phi_{i,j,k+1} D_z^0 \Phi_{i,j,k}|}{h^2 |D_z^+ \Phi_{i,j,k}| \|\nabla_0^\varepsilon \Phi_{i,j,k}|} & \text{if } \Phi_{i,j,k}\Phi_{i,j,k+1} \le 0; \\ 0 & \text{otherwise} \end{cases}$$

$$\tilde{\delta}^{(-z)}_{i,j,k} = \begin{cases} \frac{|\Phi_{i,j,k-1} D_z^0 \Phi_{i,j,k}|}{h^2 |D_z^- \Phi_{i,j,k}| \|\nabla_0^\varepsilon \Phi_{i,j,k}|} & \text{if } \Phi_{i,j,k}\Phi_{i,j,k-1} < 0; \\ 0 & \text{otherwise} \end{cases} \tag{43}$$

where $D_\alpha^+ \Phi_{i,j,k}$, $D_\alpha^- \Phi_{i,j,k}$ and $D_\alpha^0 \Phi_{i,j,k}$ are upwind, downwind and central finite difference schemes in the α-direction ($\alpha = x, y, z$), respectively, and

$$|\nabla_0^\varepsilon \Phi_{i,j,k}| = \sqrt{(D_x^0 \Phi_{i,j,k})^2 + (D_y^0 \Phi_{i,j,k})^2 + (D_z^0 \Phi_{i,j,k})^2 + \varepsilon} \text{ with } \varepsilon = 10^{-10}.$$

In our case, the function values are available only on the interface. Therefore, a modification of the above formalism is required. After some derivation, we have the following integral expression

$$\int_\Gamma f(x, y, z) dS \approx \sum_{(i,j,k) \in I} \left(f(x_o, y_j, z_k) \frac{|n_x|}{h} + f(x_i, y_o, z_k) \frac{|n_y|}{h} + f(x_i, y_j, z_o) \frac{|n_z|}{h} \right) h^3, \tag{44}$$

where (x_o, y_j, z_k) is the intersecting point of the interface and the x meshline that passes through (i, j, k), and n_x is the x component of the unit normal vector at (x_o, y_j, z_k). Similar relations exist between (x_i, y_o, z_k) and n_y, and between (x_i, y_j, z_o) and n_z, respectively. We here define irregular grid points as points with neighbor from the other side of the interface. Since Eq. (44) has already taken into account for the contribution from irregular grid points outside the interface, the summation is restricted to I, the set of irregular grid points inside or on the interface. Based on Eq. (41), the derivation of Eq. (44) is quite straightforward, and is illustrated with $\tilde{\delta}^{(+x)}_{i,j,k}$.

1. In the expression of $\tilde{\delta}^{(+x)}_{i,j,k}$, $\left|\frac{\Phi_{i+1,j,k}}{D^{+}_{x}\Phi_{i,j,k}}\right|$ and $\left|\frac{D^{0}_{x}\Phi_{i,j,k}}{\nabla^{2}_{0}\Phi_{i,j,k}}\right|$ are numerical approximations of $h^{+}_{x}=(x_{i+1}-x_{o})$ and $|n_x|$, which can be analytically evaluated since one knows the interface location and normal information of the molecular surface at its intercepting point.
2. For grid point (i, j, k) inside the interface with a negative Φ value, if its neighboring point, e.g., $(i+1, j, k)$, has a positive Φ value, the discrete delta function at (i, j, k) will have a component corresponding to $\tilde{\delta}^{(+x)}_{i,j,k}$ in Eq. (43). Meanwhile, the discrete delta function at $(i + 1, j, k)$ will have a component corresponding to $\tilde{\delta}^{(-x)}_{i+1,j,k}$ in Eq. (43). The sum of the two components at (i, j, k) and $(i + 1, j, k)$ is exactly $\frac{|n_x|}{h}$.
3. For a sharp interface, it is possible that both sides of $(i, j, k,)$ have positive Φ values, then Eq. (44) will have two terms in the x-direction at (x^{+}_{o}, y_j, z_k) and (x^{-}_{o}, y_j, z_k).
4. In Eq. (41), the density function f is evaluated at grid point (x_i, y_j, z_k). For our application, the density function is defined only on the interface, therefore we use interface values $f(x_o, y_j, z_k)$, $f(x_i, y_o, z_k)$ and $f(x_i, y_j, z_o)$ instead. In fact, the results of surface integration evaluated at the interface are better than those evaluated at grid points, according to our numerical tests.
5. Eq. (44) is also valid for the case where the interface exactly crosses a grid point.

2.5.3 The distribution of dielectric boundary forces to individual atoms—In the previous two subsections, we discuss the evaluations of numerical surface integral and dielectric boundary force. To compute dielectric forces on individual atoms, we need to distribute each unit dielectric boundary force, i.e., the force density multiplied by the associated area element, to related neighboring atoms and then collect all the forces on each specific atom. This procedure requires to classify the interfaces points into several categories since different treatments will be applied.

Fig. 2(a) shows an example of the molecular surface for a small system composed of 5 atoms. The red dots denote the centers of the probe, the "+" denote the centers of atoms, and the capital letters denote the contacting points between the atoms and the probe. The molecular surface consists of three type of facets. The contact surfaces are parts of atomic surfaces that directly contact the probe. The toric surfaces are the collection of the curves on the probe connecting two contact points, when the probe rolls over two contacted atoms, as shown by the curved face *CEDB* on the graph. The reentrant surfaces are the curved facet on the probe when it contacts more than three atoms, shown as *ABC* or *DEF* on the graph. It is worth mentioning that conventionally both the reentrant surface and toric surfaces are called an reentrant surface. Here, we differentiate them due to the need of different numerical treatments.

We first classify interface points as two types, on-atom points and off-atom points. For molecular surfaces, an on-atom point is a point that is on a contact surface while an off-atom point is a point that is on a toric surface and/or a reentrant surface. For the first type, we just need to find points which are within the distance of the atom radius from the center of an atom, and attributing it to the atom. For the second type, we need to find out which atoms the probe contacts with. Our method is to locate the probe center by a distance of the probe radius from the interface point in the surface normal direction. If the distance between an atomic center and the probe center is approximately the sum of atomic radius and probe

radius, then the probe is regarded as contacting this atom. For numerically generated MSs, a small tolerance (threshold) is used to classify interface points. In case of the MSMS surface, the results of the classification can be checked by those reported by the MSMS program [56]. Approximately, the optimal threshold is 0.013 when the MSMS density is set to 10, and 0.0013 for density 100.

The integral method described in previous section can give a force element on a small part of the molecular surface. A force element is directly attribute to an atom if the corresponding interface point is an on-atom point. However, if an interface point is on a toric or reentrant surface, we need to attribute the force element to all the contacted atoms.

For a toric surface, the normal direction of a surface element is in the same plane with directions pointed from the centers of the two atoms to the center of the probe, the distribution of the force element to two atoms can be calculated by

$$\mathbf{F_1}=\frac{\mathbf{F}\cdot\mathbf{f_1}-\mathbf{F}\cdot\mathbf{f_2}(\mathbf{f_1}\cdot\mathbf{f_2})}{1-(\mathbf{f_1}\cdot\mathbf{f_2})^2}\mathbf{f_1} \tag{45}$$

$$\mathbf{F_2}=\frac{\mathbf{F}\cdot\mathbf{f_2}-\mathbf{F}\cdot\mathbf{f_1}(\mathbf{f_2}\cdot\mathbf{f_1})}{1-(\mathbf{f_2}\cdot\mathbf{f_1})^2}\mathbf{f_2} \tag{46}$$

where **F** is the force from the surface point to the probe center, which will be distributed into two forces $\mathbf{F_1}$ and $\mathbf{F_2}$ pointing from the two atom centers to the probe center, respectively. Here, **f**, $\mathbf{f_1}$ and $\mathbf{f_2}$ are corresponding unit force vectors.

For reentrant surfaces, the probe might contact more than three atoms at a time. These situations are dealt with as the following. First, if the probe contacts only three atoms, the force vector is decomposed into three vectors pointing from each atomic center to the probe center. The magnitudes of three components are computed by solving the following linear system

$$\begin{bmatrix} f_1(1) & f_2(1) & f_3(1) \\ f_1(2) & f_2(2) & f_3(2) \\ f_1(3) & f_2(3) & f_3(3) \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \end{bmatrix} = \mathbf{F}^T, \tag{47}$$

where $f_i(j)$, $i, j = 1, 2, 3$, is the jth component of the unit force vector $\mathbf{f_i}$, F_1, F_2, and F_3 are the magnitude of the distributed force.

If the probe contacts more than three atoms, we need to distribute the force into more than three directions. However, this distribution is not unique, except that all atoms have the same distance from the probe. In the latter case, the force can be equally distributed. Otherwise, the force will be distributed to three atoms that are closest to the force element. Fig. 2(b) illustrates this scheme. Here, the interface point locates at point *O*, which belongs to the probe surface *ABCD*, i.e., the reentrant surface, when the probe contacts four atoms at points *A*, *B*, *C* and *D*. To avoid the difficulties of decomposing force vector *P O* to four directions, we choose the closest three atoms at points *B*, *C* and *D*, and distribute the vector *P O* into components *P B*, *P C* and *P D*. Our numerical experiments indicate this scheme is a very good approximation.

3 Validation

In this section, we validate our surface integration in Cartesian grids and overall force calculation scheme.

3.1 Validation of the surface integration in Cartesian grids

The overall accuracy, stability and reliability of dielectric boundary force calculations depend on the quality of PB solvers, the evaluation of six derivatives of the potential on the interface, the numerical surface integration and the distribution of forces to related atoms. The accuracy of MIBPB solvers has been extensively verified in the previous papers [70] and [25]. The same MIB algorithm is modified to rigorously evaluate the six partial derivatives at the interface. The basic numerical integration algorithm in Cartesian grids was validated in [57]. However, this algorithm is modified so that it can be applied to our situation. A major difference is that the density function values are defined and computed on the interface, instead of on irregular grid points. Moreover, there is no analytical result for the surface area of molecular surfaces in general. Therefore, we validate our Cartesian-grid surface integration algorithm by calculating molecular surface areas of sphere, two-atom, three-atom and eighteen-atom systems. Table 1 shows the validation of the surface integration with spheres of radii 2 and 4Å. Numerical results are compared with exact values. It is seen that errors essentially depend on the grid resolution. Change in the curvature does not affect the result very much. Approximately, second order convergence is observed, although the method was designed as a first order one.

For two- or multi-atomic systems, a common probe radius of 1.4Å with density 10 triangles per $Å^2$ is used in molecular surface generations except specified. Our results are compared with those of the MSMS [56]. We also plot the molecular surfaces of the diatomic, three-atom and eighteen-atom systems as illustrated in Fig. 3. Test cases conducted on these systems will be provided in the later sections. The locations and atomic van der Waals radii are $(x\ y\ z\ r_{vdW})$ = (1.500 0.4 0.4 1.6) and (−3.000 0.0 0.0 2.0) for the two-atom system, and (2.000 0.0 0.0 2.0), (0.000 −1.0 0.0 2.0) and (0.000 1.0 0.0 2.0) for the three-atom system.

Table 2 gives the results of our numerical integration. It is seen from the table that when the mesh is refined, the computed total areas converge to those of the MSMS, which validates the numerical integration carried out at the intersecting points of the interface and meshlines, and using the corresponding normal directions. However, the areas of contact surface, toric surface, and reentrant surface do not show a similar level of consistency with those of the MSMS. This is due to the fact that the MSMS surface was not analytically given but was in the triangular representation. We have used a threshold to categorize the MSMS molecular surface points into contact surface, toric surface and reentrant surface. For different cases, the best threshold may vary, which leads to errors. However, our tests indicate that the error in classification does not substantially affect our force calculation because the projection of a surface force element into atomic center directions is insensitive to the element classification.

3.2 Validation of force calculations

3.2.1 Methods of validation—The validation of electrostatic force calculations is a difficult issue because of the lack of analytical values even for small molecular systems. However, there are two ways to evaluate electrostatic forces and to examine the consistency of force calculations. In the first approach, an atom in a molecule is moved forward (or backward) along an axis by a small distance, e.g., 0.005Å, at a time. The electrostatic free energy of each configuration is evaluated and the electrostatic force corresponding to each

movement can be computed. For example, the force due to a small displacement δ in the x-direction on the ith atom centered at (x_i, y_i, z_i) can be calculated as

$$F(\Delta G)=\frac{\Delta G(x_i+\delta, y_i, z_i) - \Delta G(x_i - \delta, y_i, z_i)}{2\delta}. \quad (48)$$

The other way is to obtain electrostatic solvation forces of the same movement by summing over components

$$F(\text{Sum})=F(\text{RXN})+F(\text{DB})+F(\text{IB}) \quad (49)$$

where F(Sum) is the total force, and F(RXN), F(DB), and F(IB) are the reaction field, dielectric boundary, and ionic boundary forces, respectively. The consistency of the electrostatic forces calculated in two independent approaches, $F(\Delta G)$ and F(Sum), can be used to validate force schemes. As the accuracy of the MIBPB solvers has been extensively validated [75,70,25], the consistency in two independent force calculations provides a reliable check for the proposed method of electrostatic force evaluation.

3.2.2 Test on molecules—Here we consider a few test examples to validate our force calculation scheme. Normally, we let one atom of a molecule to move along an axis and compute the corresponding reaction field forces and dielectric boundary forces. Molecular surfaces are generated with the MSMS program at density 10, with a probe radius of 1.4Å, except for Case 1, where the molecular surface is analytical.

Case 1: Two-atom system: The two-atom system is a benchmark test for the validation of electrostatic force calculations [28,33]. Im et al. have provided a detail analysis for this system using their PBEQ, the CHARMM PB solver, with smoothed dielectric functions [33]. There is a tradeoff between accuracy and simplicity as these two methods are compared. In the present project, we provide MIBPB based test results. We have set the ionic strength to 0. The MIBPB-II is employed in our force evaluation.

Consider two atoms of radius 2.0Å. Each atom has a unit charge located at its atomic center. One atom is fixed at (−3, 0, 0) and the other moves from (−3,0,0) to (4,0,0). The reason for us to move the atom up to (4,0,0) is that at about (3.2,0,0), the MS breaks up into two pieces and admits two cusps.

Fig. 4 depicts the results generated with MIBPB at different mesh size (h = 0.21Å for (a) and (c), and h = 0.42Å for (b) and (d)). The units for the force are kcal/mol/e_c and the probe radius of 1.4Å is used to generate molecular surfaces. At different position of the center of the moving atom, the total forces calculated by differentiating electrostatic free energies Eq. (48) are given in red solid line. Meanwhile, the total forces calculated by summing of forces components Eq. (49) are given in blue solid line. As in (a) and (b), the sums of forces include the reaction field forces Eq. (20) in dashed green line and dielectric boundary forces Eq. (28) in dotted-dashed brown line, which is decomposed into the contact forces in dotted black line and reentrant forces in dotted-dashed pink line as shown in (c) and (d). From these pictures, we could draw the following conclusions. 1) Total forces calculated by two different approaches confirm each other, therefore validate the present method for this test case; 2) The reaction field forces, as the leading component of electrostatic forces, are smooth even for coarse mesh; 3) The dielectric boundary forces are composed of contact forces, toric forces and reentrant forces (=0 for this case), suffers from numerical instability,

especially when the mesh is coarse. However, in the view of an entire system, the dielectric boundary forces are important components with acceptable perturbation; 4) The pattern of both the reaction field forces and the dielectric boundary forces are consistent with the physical interpretation of this test model.

We also provided the detailed comparisons of the results from MIBPB and PBEQ, the PB solver in CHARMM as given in Fig. 5. Mesh size $h = 0.21$ is used to test these methods. The results calculated by the PBEQ with smoothed dielectric boundaries and by the MIBPB with sharp molecular surfaces are illustrated with dotted blue lines and shown with solid red lines, respectively. From Fig. 5(a), one sees a similar pattern of the solvation free energy, which confirms the reliability of two methods. We also noticed that there are about 30 kcal/mol energy difference in two results, which is due to the fact that the PBEQ uses a smoothed dielectric interface as discussed by [60,61,63].

The total forces obtained by the force density approaches are given in Figs. 5(b) for the PBEQ and the MIBPB. The force curve from the MIBPB is almost completely smooth and monotonically decay. The smoothness indicates that the present method is highly stable. The monotonically decay in the total force curve is consistent with underlying physics of two separating charged spheres.

Fig. 4(c) gives a comparison of the reaction field forces obtained with two methods. These forces obtained from two methods have similar patterns. However, the peak force value of the PBEQ is about 40% higher than that of the MIBPB. This is consistent with what reported in the literature that the smoothed dielectric boundaries result in 10–40% over-estimation of solvation energies and forces [63, 60].

It is difficult to obtain stable dielectric boundary forces as shown in Fig. 5. Physically, we expect a smooth decay of the dielectric boundary forces as two atoms separate from each other. The dielectric boundary forces of the MIBPB shown in Fig. 5(d) have a monotonically decaying pattern, which leads to the same pattern in the total force calculation. The dielectric boundary forces of the PBEQ shown in Fig. 5(d) slightly undulate over the coordinate from −3 to 1.

Case 2: Three-atom system: Since molecular surfaces admit reentrant surfaces, it is important to further validate our algorithm for a 3-atom system. In this case, one atom is moved from (−2,0,0) to (2,0,0) in the x-direction while the other two atoms are fixed at (0,−1,0) and (0,1,0), respectively. All atoms have radius 2.0Å and charge $1.0e_c$. From Fig. 6-(a), it is seen a good consistency of the x-direction forces on the moving atom obtained by two different methods. The forces of contact surfaces (F(Contact)) are separated from those of toric and reentrant surfaces (F(Toric+Reen)). The symmetric pattern about $x = 0$ confirms the symmetric setup of the testing system. It is worth mentioning that the forces calculated by electrostatic free energies show more oscillations, which in fact largely depend on the quality of the MSMS surface generation. These oscillations can be observed with different settings of the MSMS parameters. In contrast, the total forces calculated by the summation of force components are much smoother and more stable. This is important for PB based molecular dynamics simulations.

Case 3: Eighteen-atom system: As seen in Fig. 3, the eighteen-atom system, Cyclohexane, is composed of 6 carbon atoms and 12 hydrogen atoms. The partial charges of these atoms are artificially assigned to the atomic centers as $1\ e_c$. In this test, we fixed the coordinates of atom 2–18 while move the first one, a carbon atom located at (0.9540 −0.6510 1.7), in the x-direction forward and backward 1 Å. In Fig. 6-(b), electrostatic forces on the first atom obtained by the two different methods show a good consistency.

Case 4: A biomolecule example: a Low-density lipoprotein receptor (PDB: 1ajj): The final example is chosen as the molecular surface of a Low-density lipoprotein receptor (PDB: 1ajj) with 37 residues and 519 atoms. Fig. 7(a) shows its structure. We move the outmost nitrogen natively located at (−0.169, 7.698, 13.415) along the positive and negative direction of the *x*-axis for a distance of 0.25Å. This movement cannot be too large, or it will violate the physical limitation. The calculation results are shown in Fig. 7(b) where the confirmation of results from two methods has been achieved.

3.2.3 Force correlation test—Finally, we provide an alternative way to confirm the convergence of the resulting forces from our method. In Fig. 8, we plot the correlation of forces obtained at different mesh sizes, i.e., 0.25Å, 0.5 Å and 1.0 Å, for 1ajj. All the forces in *x*, *y* and *z* directions for each atom are plot here. From the figure, we can see a good convergence of reaction field forces at mesh sizes of 1.0Å and 0.5 Å compared with that obtained at 0.25 Å. For dielectric boundary forces, as well as their toric and reentrant components, their accuracy is significantly reduced at the mesh size of 1.0Å, because of the error in the surface integration conducted at such a coarse mesh. However, at 0.5Å, the convergence is good as seen in Figs. 8(c)–(e).

4 Application to molecular dynamics simulations

4.1 The implementation of MIBPB based PB forces in the AMBER package

The present MIBPB calculates the electrostatic potential and provides electrostatic forces asserted on each individual atoms for solvated biomolecules. However, to perform the molecular dynamics simulation, it takes more computational modules such as other components of the force field, initialization, randomness, time evolution and so on. In this work, we implement the MIBPB in the AMBER package. Specifically, we replace the original PB solver and force driver in the AMBER [43,45] with our MIBPB solver and MIBPB force driver, respectively. We use *igb* = 8, a value currently not being used by AMBER-GB modules, to switch the implicit solvent model to the MIBPB. We modified a few files and included all the necessary files of MIBPB in the sander package of AMBER. We plan to contact the authors of AMBER/CHARMM to see if they would officially contain our package after our method becomes more stable and efficient.

It is also necessary for us to point out that the MIBPB, as still in the early stage of its development, does not provide many options for the user. The molecular surface is obtained by running the MSMS [56]. Charges are treated by trilinear interpolations which distribute the charge at a location to its closest eight grid points with weights decided by the distance between the original charge location and the grid point. The FFT is used to solve the PBE in vacuum, and a preconditioned biconjugate method is used to solve PBE in solution.

In summary, to run the MIBPB module embedded in the AMBER, we change the algorithm in providing reaction field forces, dielectric boundary forces, ionic boundary forces, and electrostatic solvation energy while keep the force fields of bond, angle, dihedral, van der Waals, Coulomb and all others intact.

4.2 Molecular dynamics simulations

In this subsection, we demonstrate MD simulations using the MIBPB based PB approach in conjugation with the AMBER package. Three biomolecular systems are considered in our study. To ensure the reliability of the PB results, all simulations are conducted at the mesh size of 0.3Å. Time increment is set to 10^{-15}s. All MD data, including the configuration for computing RMSDs, are collected at each 50 time steps.

Case 1: Alanine dipeptide—The first system is the alanine dipeptide with 22 atoms, which was studied by Im et al [33]. As suggested by the AMBER tutorial, we started from a pdb file with the first atoms (C,C,N) of the three residue/terminal (ACE, ALA, NME), and used *leap*, one of the functional modules of the AMBER to generate parameters and structure, and to add other 19 atoms. We relaxed the system with the graphic tools provided by *leap* and minimized it with 200 steps of steepest descent runs and 300 steps of conjugate gradient runs. After that, we run 300ps MD (i.e., 300,000 steps) simulations. We use the AMBER option to remove translational and rotational motions after every 500 steps. The results are given in Fig. 9. From the results, it is seen that the RMSD reduces after some initial steps but keeps certain level of irregular oscillations, which are due to the small size of the system. And later, with increased size of the system, we will seen more and more stable results.

Case 2: WW domain binding protein-1—The other biomolecular system used is the WW domain binding protein-1, the peptide bound to 65 KDA yes-associated protein. The structure is obtained from the complex (1jmq) and has 10 residues (GTPPP-PYTVG), 138 atoms. We follow the AMBER procedure to obtain the pdb file and add hydrogens by using the *leap*. The minimization and MD simulation have been run in the same setup as that of the first system. The translational and rotational motions are removed after every 100 steps. From Fig. 10, one can see that RMSDs shown in (a) and temperature shown in (d) are more stable due to the increased size of the system. Most of ten residues in the chain are non-polar, and a few of them are polar. There is no charged residue in the system. This kind of polymer chains in the water will tend to fold due to the hydrophobicity, as confirmed by the trajectory in VMD. It is found that structures obtained by the MIBPB based MD simulations stay mostly in folded morphologies. We further verified our conjecture by checking the averaged distances between all C_α atoms at the end of each 50 steps are plotted in Fig. 11(a). The MIBPB based MD method predicts smaller C_α distances with time until convergence, indicating a more compact packing of residues.

Case 3: Trp-cage miniprotein (1L2Y)—We further consider a Trp-cage miniprotein (1L2Y) which has 95% folded structures in water at physiological pH [48]. This molecule has 20 residues and 340 atoms, representing a novel fold, designated as the ‘Trp-cage’ motif. It is known to be significantly more stable than other miniproteins reported [48]. The folding of this miniprotein is synergic and is hydrophobically driven by the encapsulation of a Trp side chain in a sheath of Pro rings. As a small protein-like molecule, this Trp-cage miniprotein was advocated as a benchmark test for both experimental studies and computational predictions of protein folding and unfolding pathways [48].

Our simulations were carried out in the same procedure as described in the last case. As shown in Fig. 12, the biomolecule approaches to stabilized solvation energy and RMSD. The averaged distances between all C_α atoms at the end of each 50 steps are plotted in Fig. 11(b), which confirms the folding and stabilizing procedure. Typical structures obtained from NMR experiments, as the initial structure of MD simulation and that from the MIBPB based MD simulation are also illustrated in Fig. 13. The molecule folded after the MD simulation and this can be associated with the polarity and charge distribution of residues of the molecule.

Finally, we point out that compared with the AMBER PB module, the present MIBPB based MD method is a few times slower. There are a few reasons that hinder the efficiency of the present method. First, at each time step, MIBPB based MD simulations require the generation of the molecular surface by the MSMS, which is the bottleneck at present. Additionally, AMBER typically has a seven-diagonal matrix, whereas to better resolve jump singularity and geometric singularity, the MIBPB based PB solver has a more complicated

matrix structure. Furthermore, the force computations in MIBPB based MD simulations use more complicated algorithms. However, in this paper, we have provided a sufficient evidence to demonstrate the advantage in accuracy and stability of the current method. We hope this could arouse the attention from the mathematical community to further develop powerful mathematical algorithms to improve the efficiency of implicit solvent models.

5 Concluding remarks

Multiscale implicit solvent models have attracted much attention in the past decade due to its efficiency in reducing computational cost by modeling the solvent and mobile ions as continuum. The Poisson-Boltzmann (PB) equation is a primary implicit solvent model and has been widely applied in molecular biology and material sciences. PB based molecular dynamics (MD) methods have a potential to tackle large biomolecules and have been extensively studied in the past few decades. However, no interface based technique has been applied to any of the existing PB based MD methods. The present work presents the first development of an interface technique based MD method via the PB formalism. The matched interface and boundary (MIB) method [68,69,72,73,74] is developed to handle the sharp molecular surface (MS) in the PB based MD approach. This work is a natural extension of our previous endeavor on the development of MIB based PB solvers [25,70,75].

An MIB based PB solver, MIBPB-II [70], is employed to obtain electrostatic potentials. To calculate the electrostatic forces associated with the use of sharp MSs, new force density expressions are derived. To compute dielectric boundary forces, a complete set of potential gradients are required at all the intersecting points of the interface and meshlines. These gradients are calculated by extending our earlier MIBPB scheme. The surface integration in the Cartesian grid is a crucial issue in the dielectric boundary force calculation that often limits the accuracy and convergence. To this end, an advanced surface integration technique developed by Smereka [57] is adopted with a minor modification. Unlike the van der Waals surface where each facet belongs uniquely to an atom, the MS admits reentrant surfaces which do not uniquely belong to a single atom. Therefore, forces from the reentrant surfaces have to be assigned to related atoms. A set of force distribution schemes are introduced. The resulting electrostatic forces due to the displacements of atoms are carefully validated by a comparison with those obtained by differentiating the electrostatic free energy. A large test set, including a diatomic system, a triatomic system, a cyclohexane, and a low-density lipoprotein receptor, is utilized in our validation. Correlations of results obtained at different mesh sizes are plotted to verify the convergence of the present method. The correlation patterns indicate good convergence of reaction field forces at mesh sizes as coarse as 1.0Å. For dielectric boundary forces, good convergence can be obtained at the mesh size of near 0.5Å. The MIBPB solver and its force calculation are then interfaced with the AMBER MD package, Sander, parallelizing to its original finite difference based PB solver [43,45].

As our first attempt, the present MIBPB based MD package is far from mature and can be improved in a number of ways. First, the MIBPB-III method [25] which employs the Green's function to alleviate charge singularity should be able to provide accurate electrostatic potentials at much coarser meshes. Additionally, the current numerical surface integral scheme is of first order in convergence and can be improved. The accuracy of dielectric boundary forces is a limiting fact at large grid sizes in the present method. Moreover, MSs are generated in triangular meshes by using the MSMS program [56] and converted to Cartesian meshes by a scheme developed in our earlier work [75]. This setup is normally stable during the MD simulation but does cause a stability problem on a very rare basis. Furthermore, the preconditioned bi-conjugate solver is used in solving PB equation and its matrix system has been optimized in MIBPB-II. However, the choice of a

precondition matrix and even the decision about which linear system solver is to be used is still worthy of further exploration. Finally, the most time-consuming part is the molecular surface generation. There is a pressing need to speed up the molecular surface construction in the Cartesian domain. Due to these reasons, the present MIBPB based MD package can only meet the speed requirement of small-scale systems in relatively short simulation time.

Acknowledgments

This work was supported in part by NSF grants DMS-0616704 and CCF-0936830, and NIH grants CA-127189 and GM-090208. The authors thank Nathan Baker, Wonpil Im, Ray Luo, Peter Smereka and Yongcheng Zhou for useful discussions. Valuable suggestions from anonymous referees are also acknowledged.

References

1. Babuška I. The finite element method for elliptic equations with discontinuous coefficients. Computing 1970;5:207–213.
2. Baker N, Holst M, Wang F. Adaptive Multilevel Finite Element Solution of the Poisson-Boltzmann Equation II. Refinement at Solvent-Accessible Surfaces in Biomolecular Systems. J Comp Chem 2000;21:1343–1352.
3. Baker NA, Sept D, Joseph S, Holst MJ, McCammon JA. Electrostatics of nanosystems: application to microtubules and the ribosome. PNAS 2001;98:10037–10041. [PubMed: 11517324]
4. Baker NA, Sept D, Holst MJ, McCammon JA. The adaptive multilevel finite element solution of the Poisson-Boltzmann equation on massively parallel computers. IBM J Res Dev 2001;45:427–438.
5. Baker NA. Improving implicit solvent simulations: a Poisson-centric view. Current Opin Struct Biol 2005;15:137–143.
6. Bates P, Wei GW, Zhao S. Minimal molecular surfaces and their applications. J Comput Chem 2008;29:380–391. [PubMed: 17591718]
7. Bates P, Chen Z, Sun YH, Wei GW, Zhao S. Geometric and potentialdriving formation and evolution of biomolecular surfaces. J Math Biol 2009;59:193–231. [PubMed: 18941751]
8. Bordner A, Huber G. Boundary element solution of the linear Poisson–Boltzmann equation and a multipole method for the rapid calculation of forces on macromolecules in solution. J Comput Chem 2003;24:353–367. [PubMed: 12548727]
9. Boschitsch A, Fenley M, Zhou HX. Fast boundary element method for the linear Poisson–Boltzmann equation. J Phys Chem, B 2002;106:2741–2754.
10. Boschitsch A, Fenley M. Hybrid boundary element and finite difference method for solving the nonlinear Poisson–Boltzmann equation. J Comput Chem 2004;25:935–955. [PubMed: 15027106]
11. Bramble J, King J. A finite element method for interface problems in domains with smooth boundaries and interfaces. Adv Comput Math 1996;6:109–138.
12. Brooks BR, Bruccoleri RE, Olafson BD, States DJ, Swaminathan S, Karplus M. CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. J Comput Chem 1983;4:187–217.
13. Case DA, Cheatham TE III, Darden T, Gohlke H, Luo R, Merz KM Jr, Onufriev A, Simmerling C, Wang B, Woods R. The Amber biomolecular simulation programs. J Computat Chem 2005;26:1668–1688.
14. Chen, Zhan; Baker, Nathan; Wei, GW. Differential geometry based solvation model I: Eulerian formulation. Journal of Computational Physics. 2010 in press, (40 pages).
15. Chern IL, Shu YC. Coupling interface method for elliptic interface problems. Journal of Computational Physics 2007;225:2138–2174.
16. Cortis CM, Friesner RA. Numerical solution of the Poisson-Boltzmann equation using tetrahedral finite-element meshes. J Comput Chem 1997;18:1591–1608.
17. Connolly ML. Molecular surface triangulation. J Appl Crystallogr 1985;18:499–505.
18. Davis ME, Madura JD, Luty BA, McCammon JA. Electrostatics and diffusion of molecules in solution: simulations with the University of Houston Brownian dynamics program. Comput Phys Commun 1991;62:187–197.

19. Eisenhaber F, Argos P. Improved strategy in analytic surface calculation for molecular systems: Handling of singularities and computational efficiency. J Comput Chem 1993;14:1272–1280.
20. Fedkiw RP, Aslam T, Merriman B, Osher S. A non-oscillatory Eulerian approach to interfaces in multimaterial flows (the ghost fluid method). J Comput Phys 1999;152:457–492.
21. Feig M, Brooks CL III. Recent advances in the development and application of implicit solvent models in biomolecule simulations. Current Opinions in Struct Bio 2004;14:217–224.
22. Feig M, Onufriev A, Lee MS, Im W, Case DA, Brooks CL III. Performance comparison of generalized Born and Poisson methods in the calculation of electrostatic solvation energies for protein structures. J Comp Chem 2004;25:265–284. [PubMed: 14648625]
23. Forsten KE, Kozack RE, Lauffenburger DA, Subramaniam S. Numerical solution of the nonlinear Poisson-Boltzmann equation for a membrane-electrolyte system. J Phys Chem 1994;98:5580–5586.
24. Fogolari F, Brigo A, Molinari H. Protocol for MM/PBSA molecular dynamics simulations of proteins. Biophys J 2003;85:159–166. [PubMed: 12829472]
25. Geng WH, Yu SN, Wei GW. Treatment of charge singularities in the implicit solvent models. J Chem Phys 2007;128:114106. [PubMed: 17887827]
26. Georgescu RE, Alexov EG, Gunner MR. Combining conformational flexibility and continuum electrostatics for calculating pKas in proteins. Biophys J 2002;83:1731–1748. [PubMed: 12324397]
27. Grant JA, Pickup BT, Nicholls A. A smooth permittivity function for Poisson-Boltzmann solvation methods. Journal of Computational Chemistry 2001;22:608–40.
28. Gilson MK, Davis ME, Luty BA, McCammon JA. Computation of electrostatic forces on solvated molecules using the Poisson-Boltzmann Equation. J Phys Chem 1993;97:3591–3600.
29. Gogonea V, Osawa E. Implementation of solvent effect in molecular mechanics. 1. Model development and analytical algorithm for the solvent - Accessible surface area. Supramol Chem 1994;3:303–317.
30. Holst M, Saied F. Multigrid solution of the Poisson-Boltzmann equation. J Comput Chem 1993;14:105–113.
31. Holst M, Baker N, Wang F. Adaptive multilevel finite element solution of the Poisson-Boltzmann equation I. Algorithms and examples. J Comput Chem 2000;21:1319–1342.
32. Honig B, Nicholls A. Classical electrostatics in biology and chemstry. Science 1995;268:1144–1149. [PubMed: 7761829]
33. Im, Wonpil; Beglov, Dmitrii; Roux, Benoit. Continuum solvation model: computation of electrostatic forces from numerical solutions to the Poisson-Boltzmann equation. Computer Physics Communications 1998;111:59–75.
34. Juffer A, Botta E, van Keulen B, van der Ploeg A, Berendsen H. The electric potential of a macromolecule in a solvent: a fundamental approach. J Comput Phys 1991;97:144–171.
35. Klapper I, Hagstrom R, Fine R, Sharp K, Honig B. Focusing of electric fields in the active site of Cu-Zn superoxide dismutase: Effects of ionic strength and amino-acid modification. Proteins 1986;1:47–59. [PubMed: 3449851]
36. Kollman PA, Massova I, Reyes C, Kuhn B, Huo S, Chong L, Lee M, Lee T, Duan Y, Wang W. Calculating structures and free energies of complex molecules: combining molecular mechanics and continuum models. Acc Chem Res 2000;33:889–897. [PubMed: 11123888]
37. Lee B, Richards FM. Interpretation of protein structures - estimation of static accessibility. J Mol Biol 1971;55:379. [PubMed: 5551392]
38. LeVeque RJ, Li ZL. The immersed interface method for elliptic equations with discontinuous coefficients and singular sources. SIAM J Numer Anal 1994;31:1019–1044.
39. Liang J, Subranmaniam S. Computation of molecular electrostatics with boundary element methods. Biophys J 1997;73:1830–1841. [PubMed: 9336178]
40. Liu WK, Liu Y, Farrell D, Zhang L, Wang X, Fukui Y, Patankar N, Zhang Y, Bajaj C, Chen X, Hsu H. Immersed finite element method and its applications to biological systems. Computer Methods in Applied Mechanics and Engineering 2006;195:1722–1749. [PubMed: 20200602]

41. Liu XD, Fedkiw RP, Kang M. A boundary condition capturing method for Poisson's equation on irregular domains. J Comput Phys 2000;160:151–178.
42. Lu BZ, Chen WZ, Wang CX, Xu XJ. Protein molecular dynamics determined by a single Poisson-Boltzmann calculation. Proteins: Struct Func Genet 2002;48:497–504.
43. Lu Q, Luo R. A Poisson-Boltzmann dynamics method with nonperiodic boundary condition. J Chem Phys 2003;119:11035–11047.
44. Lu BZ, Cheng XL, Huang JF, McCammon JA. Order N algorithm for computation of electrostatic interactions in biomolecular systems. PNAS 2006;103:19314–19319. [PubMed: 17148613]
45. Luo R, David L, Gilson MK. Accelerated Poisson-Boltzmann calculations for static and dynamic systems. J Comput Chem 2002;23:1244–1253. [PubMed: 12210150]
46. MacKerell AD jr, Bashford D, Bellott M, Dunbrack JD, Evanseck MJ, Field MJ, Fischer S, Gao J, Guo H, Ha S, Joseph-McCarthy D, Kuchnir L, Kuczera K, Lau FTK, Mattos C, Michnick S, Ngo T, Nguyen DT, Prodhom B, Reiher WE, Roux B, Schlenkrich M, Smith JC, Stote R, Straub J, Watanabe M, Wiorkiewicz-Kuczera J, Yin D, Karplus M. All-atom empirical potential for molecular modeling and dynamics studies of proteins. J Phys Chem 1998;102:3586–3616.
47. Mayo A. The fast solution of Poisson's and the biharmonic equations on irregular regions. SIAM J Numer Anal 1984;21:285–299.
48. Neidigh JW, Fesinmeyer RM, Andersen NH. Designing a 20-residue protein. Nat Struct Biol 2002;9:425–430. [PubMed: 11979279]
49. Nielsen JE, McCammon JA. On the evaluation and optimization of protein X-ray structures for pKa calculations. Protein Sci 2003;12:313–326. [PubMed: 12538895]
50. Prabhu NV, Zhu P, Sharp KA. Implementation and testing of stable, fast implicit solvation in molecular dynamics using the smooth-permittivity finite difference Poisson-Boltzmann method. J Comput Chem 2004;25:2049–2064. [PubMed: 15481091]
51. Peskin CS. Numerical analysis of blood flow in heart. J Comput Phys 1977;25:220–252.
52. Qiao ZH, Li ZL, Tang T. A finite difference scheme for solving the nonlinear Poisson-Boltzmann equation modeling charged spheres. J Comp Math 2006;24:252–264.
53. Richards FM. Areas, volumes, packing and protein structure. Annu Rev Biophys Bioeng 1977;6:151–176. [PubMed: 326146]
54. Rocchia W, Alexov E, Honig B. Extending the applicability of the nonlinear Poisson-Boltzmann equation: multiple dielectric constants and multivalent ions. J Phys Chem B 2001;105:6507–6514.
55. Roux B, Simonson T. Implicit solvent models. Biophys Chem 1999;78:1–20. [PubMed: 17030302]
56. Sanner MF, Olson AJ, Spehner JC. Reduced surface: An efficient way to compute molecular surfaces. Biopolymers 1996;38:305–320. [PubMed: 8906967]
57. Smereka P. The numerical approximation of a delta function with application to level set methods. J Comput Phys 2006;211:77–90.
58. Sharp KA, Honig B. Electrostatic interactions in macromolecules: thoery and applications. Annu Rev Biophys Biophys Chem 1990;19:301–332. [PubMed: 2194479]
59. Swanson JMJ, Henchman RH, McCammon JA. Revisiting free energy calculations: a theoretical connection to MM/PBSA and direct calculation of the association free energy. Biophys J 2004;86:67–74. [PubMed: 14695250]
60. Swanson JMJ, Mongan J, McCammon JA. Limitations of atom-centered dielectric functions in implicit solvent models. J Phys Chem 2005;109:14769–14772.
61. Swanson JMJ, Wagoner JA, Baker NA, McCammon JA. Optimizing the Poisson dielectric boundary with explicit solvent forces and energies: lessons learned with atom-centered dielectric functions. J Chem Theory Comput 2007;3:170–183.
62. Vorobjev YN, Scheraga HA. A fast adaptive multigrid boundary element method for macromolecular electrostatic computations in a solvent. J Comput Chem 1997;18:569–583.
63. Wagoner JA, Baker NA. Solvation Forces on Biomolecular Structures: A Comparison of Explicit Solvent and Poisson-Boltzmann Models. J Comput Chem 2004;25:1623–1629. [PubMed: 15264256]
64. Warwicker J. Improved pKa calculations through flexibility based sampling of a water-dominated interaction scheme. Protein Sci 2004;13:2793–2805. [PubMed: 15388865]

65. Warwicker J, Watson HC. Calculation of the electric-potential in the active-site cleft due to alpha-helix dipoles. J Mol Biol 1982;154:671–679. [PubMed: 6288964]
66. Wei GW. Differential geometry based multiscale models. Bulletin of Mathematical Biology 2010;72:1562–1622. [PubMed: 20169418]
67. Xie D, Zhou SZ. A new minimization protocol for solving nonlinear Poisson-Boltzmann mortar finite element equation. BIT Numerical Mathematics 2007;47:853–871.
68. Yu SN, Zhou YC, Wei GW. Matched interface and boundary (MIB) method for elliptic problems with sharp-edged interfaces. J Comput Phys 2007;224:729–756.
69. Yu SN, Wei GW. Three-dimensional matched interface and boundary (MIB) method for treating geometric singularities. J Comput Phys 2007;227:602–632.
70. Yu SN, Geng WH, Wei GW. Treatment of geometric singularities in the implicit solvent models. J Chem Phys 2007;126:244108. [PubMed: 17614538]
71. Zauhar RJ, Morgan RS. A new method for computing the macromolecular electric-potential. J Mol Biol 1985;186:815–820. [PubMed: 4093987]
72. Zhao S, Wei GW. High order FDTD methods via derivative matching for Maxwell's equations with material interfaces. J Comput Phys 2004;200:60–103.
73. Zhou YC, Zhao S, Feig M, Wei GW. High order matched interface and boundary (MIB) schemes for elliptic equations with discontinuous coefficients and singular sources. J Comput Phys 2006;213:1–30.
74. Zhou YC, Wei GW. On the fictitious-domain and interpolation formulations of the matched interface and boundary (MIB) method. J Comput Phys 2006;219:228–246.
75. Zhou YC, Feig M, Wei GW. Highly accurate biomolecular electrostatics in continuum dielectric environments. J Comput Chem 2008;29:87–97. [PubMed: 17508411]
76. Zhang Y, Xu G, Bajaj C. Quality meshing of implicit solvation models of biomolecular structures. Computer Aided Geometric Design 2006;23:510–530. [PubMed: 19809581]

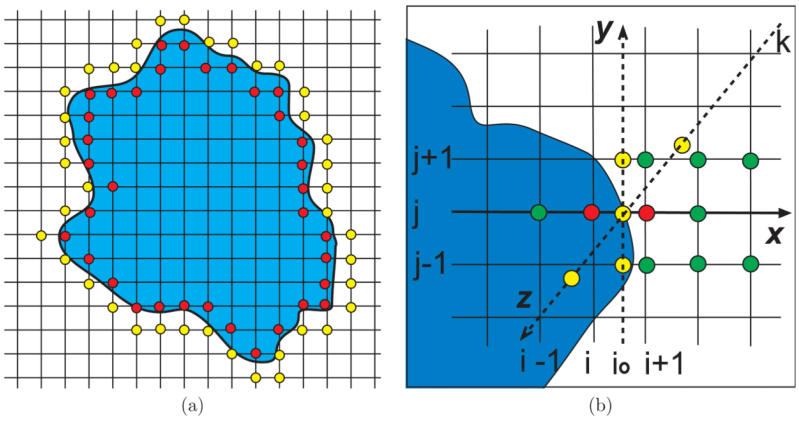

Figure 1.

An illustration of MIB schemes. The horizontal and vertical axes are the *x* and *y* directions, respectively. (a) The grid extension near the interface in a second order scheme; (b) An intersecting point at (i_o, *j*, *k*)

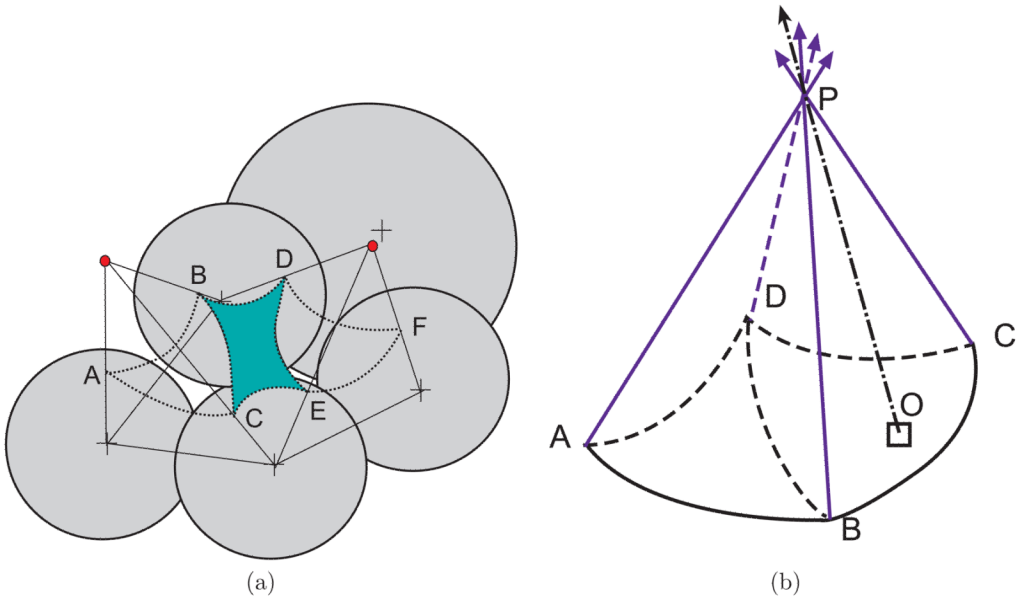

Figure 2.
Dielectric boundary force calculation. (a) Illustration of reentrant surfaces, e.g. ABC and DEF, and toric surfaces, e.g. CEDB in green, on the molecular surface; (b) Force distribution from a reentrant surface to related atoms.

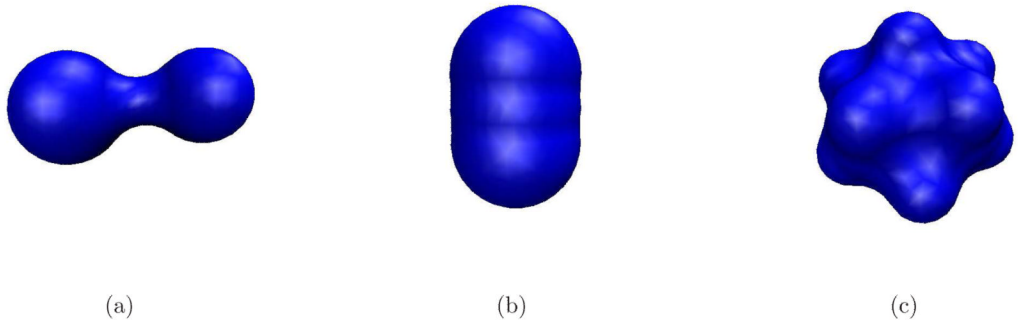

Figure 3.
Molecular Surface generated from MSMS for (a) a diatomic system (b) a three-atom system; and (c) Cyclohexane (18 atoms).

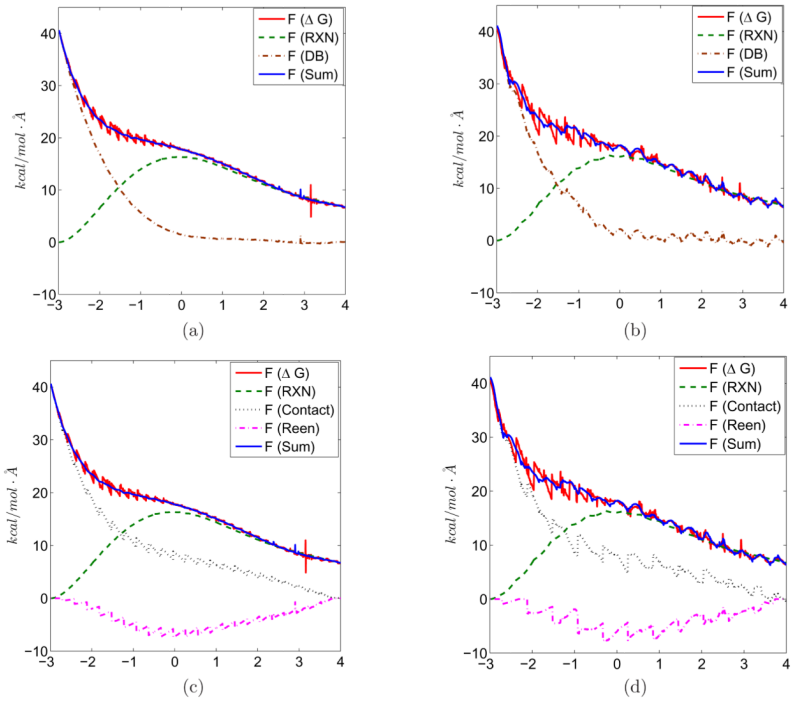

Figure 4.

Electrostatic forces calculated for two separating charged atoms. The horizontal axis is the position of the moving atom in unit Å and the vertical axis is the forces. (a) h = 0.21Å with summed dielectric boundary forces; (b) h = 0.42Å with summed dielectric boundary forces; (c) h = 0.21Å with decomposed dielectric boundary forces; (d) h = 0.42Å with decomposed dielectric boundary forces.

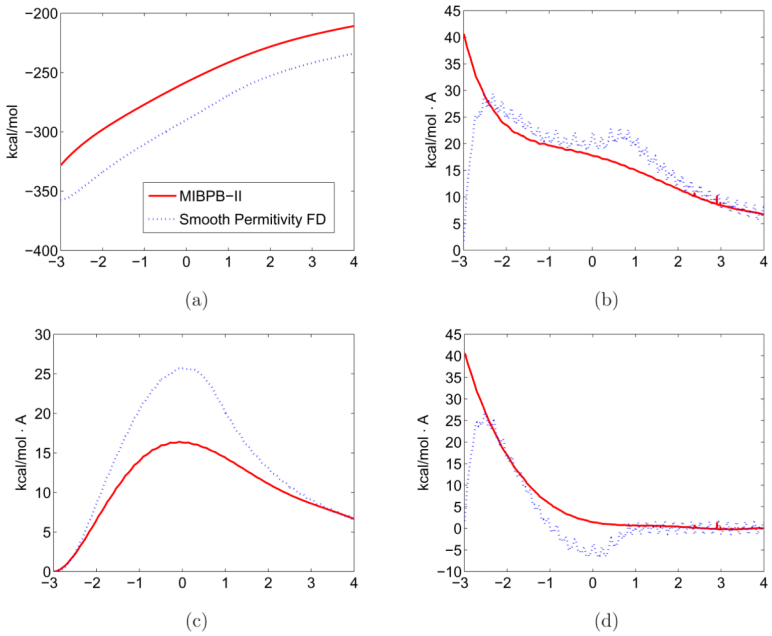

Figure 5.

Comparison of electrostatic forces on the moving atom of a two-atom systems obtained at h = 0.21Å. The horizontal axis is the position of the moving atom in unit Å. Red solid line: present MIB based method with molecular surfaces; Blue dotted line: PBEQ with smooth interface. (a) Solvation free energy of the system; (b) Total force on the moving atom calculated from the summation of reaction force and dielectric boundary force; (c) Reaction force; (d) Dielectric boundary forces;

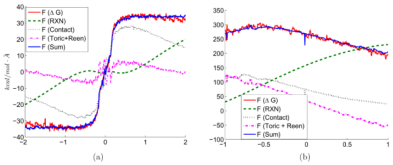

Figure 6.

Forces calculated for (a) a three-atom system and (b) Cyclohexane. The horizontal axis is the position of the moving atom in unit Å and the vertical axis is the forces.

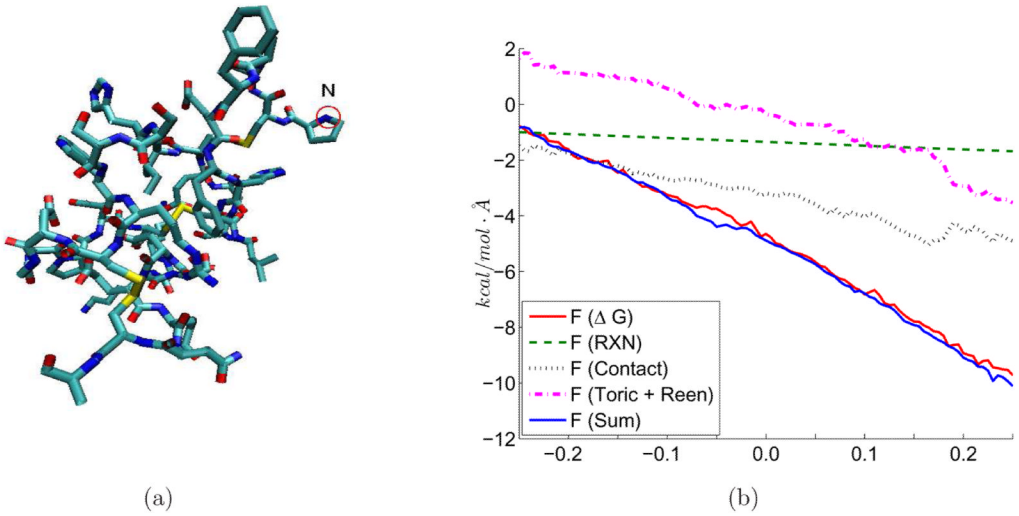

Figure 7.

A low-density lipoprotein receptor 1ajj: (a) Illustration of the molecule and the atom that is moved; (b) Electrostatic forces. The horizontal axis is the position of the moving atom in unit Å and the vertical axis is the forces.

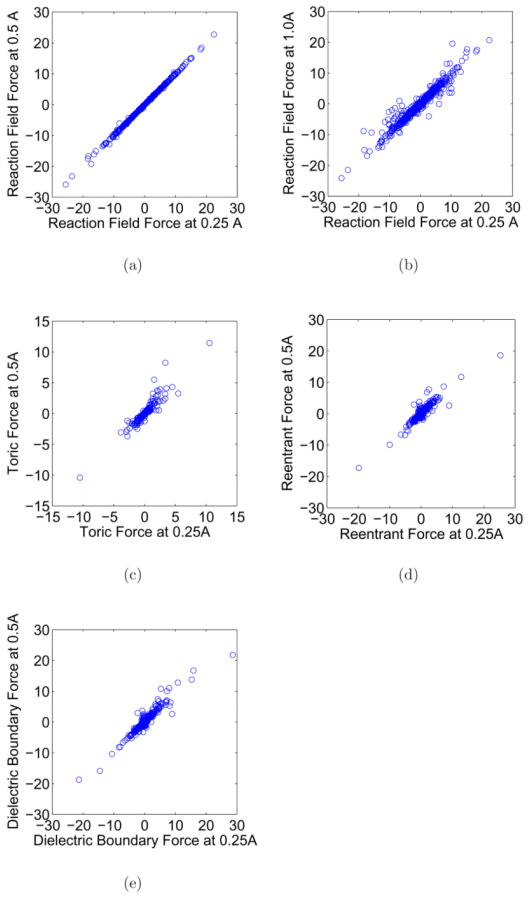

Figure 8.
Correlations of forces obtained at different mesh sizes by the MIBPB for protein 1ajj. (a) Reaction field forces between mesh sizes of 0.25 and 0.5Å; (b) Reaction field forces between mesh sizes of 0.25 and 1.0Å; (c) Toric forces between mesh sizes of 0.25 and 0.5 Å, (d) Reentrant forces between mesh sizes of 0.25 and 0.5Å, (e) Dielectric boundary forces between mesh sizes of 0.25 and 0.5Å.

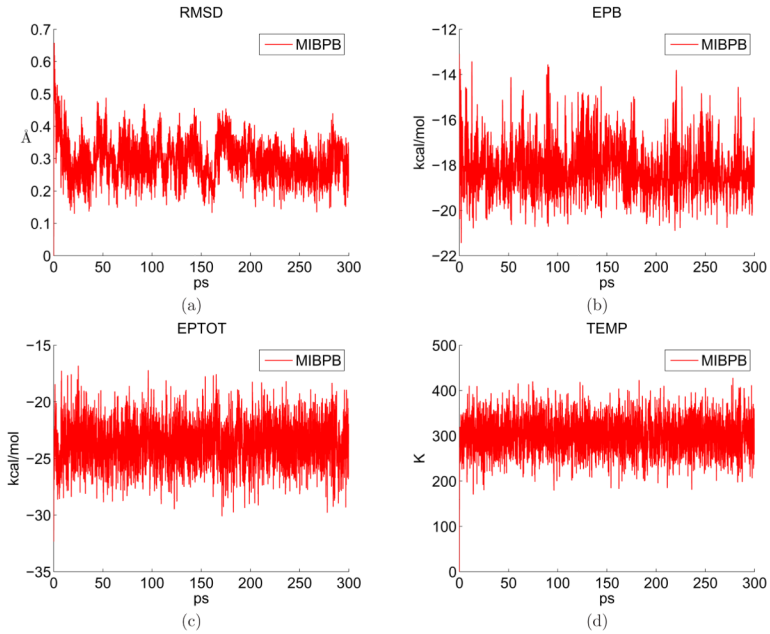

Figure 9.

300 ps MD simulations of the alanine dipeptide obtained with MIBPB based methods. The horizontal axis is time in unit of picosecond (ps). (a) RMSD; (b) Solvation free energies; (c) Total potential energies; (d) Temperature.

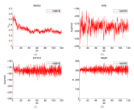

Figure 10.
A 120 ps MD simulation of the WW domain binding protein-1 obtained with the MIBPB based PB method. The horizontal axis is time in unit of picosecond (ps). (a) RMSD; (b) Solvation free energy; (c) Total potential energy; (d) Temperature.

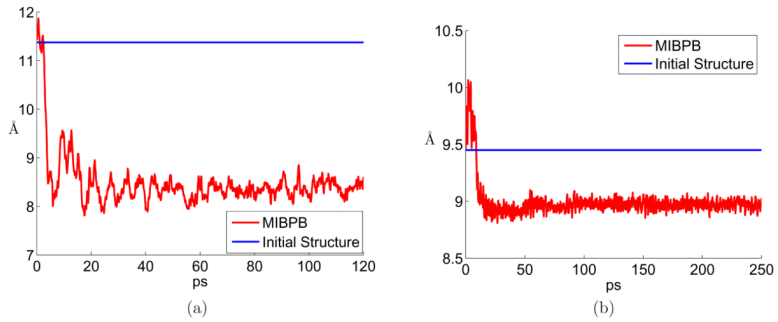

Figure 11.

A comparison of the averaged C_α atom distances. The horizontal axis is time in unit of picosecond (ps) and the vertical axis is distance. The red line is the simulation result and the blue line is that from NMR data. Results of the MIBPB are illustrated in red lines. (a) The averaged C_α atom distances for WW domain binding protein-1 (Case 2); (b) The averaged C_α atom distances for Trp-cage miniprotein (Case 3).

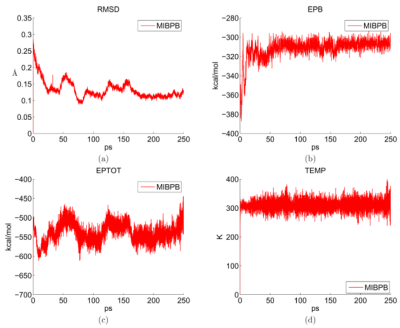

Figure 12.

250 ps MD simulations of the Trp-cage miniprotein (1L2Y) obtained with the MIBPB method. The horizontal axis is time in unit of picosecond (ps). (a) RMSD; (b) Solvation free energy; (c) Total potential energy; (d) Temperature.

(a) (b)

Figure 13.
A comparison of typical Trp-cage miniprotein structures obtained from NMR experiments and from MIBPB MD simulations. (a) Obtained from NMR experiments as the initialization structure; (b) Obtained from the MIBPB based MD simulation as the final structure.

Table 1

Integration errors for spherical surfaces

Mesh	Radius 2Å			Radius 4 Å		
h	Area	Error	Order	Area	Error	Order
1	41.77	8.50	-	192.94	8.12	-
0.5	48.23	2.03	2.07	198.67	2.40	1.76
0.25	49.67	0.60	1.76	200.58	0.49	2.30
0.125	50.14	0.12	2.30	200.92	0.14	1.78
0.0625	50.23	0.04	1.78	201.03	0.03	1.95
Exact	50.265			201.062		

Table 2

Comparison of surface areas calculated from numerical integrations and from the MSMS

	Two-atom				Three-atom				18-atom			
h	**total**	**cont.**	**toric**	**reen**	**total**	**cont.**	**toric**	**reen**	**total**	**cont.**	**toric**	**reen**
0.5	82.73	70.08	12.46	0.00	90.85	76.01	13.35	0.99	132.05	64.16	51.56	15.99
0.25	83.82	70.79	12.98	0.00	92.88	76.76	14.88	1.11	131.01	65.50	49.13	16.30
0.125	84.32	71.57	12.73	0.00	93.92	78.30	14.39	1.19	130.99	65.76	48.73	16.41
0.0625	84.44	71.63	12.84	0.00	93.98	77.94	14.73	1.21	131.16	65.46	48.71	16.91
MSMS	84.41	70.71	13.70	0.00	94.03	75.61	17.68	0.74	131.32	64.17	52.61	14.32

J Comput Phys. Author manuscript; available in PMC 2012 January 20.