Treatment of charge singularities in implicit solvent models

Weihua Geng, Sining Yu, and Guowei Wei[a)]
Department of Mathematics, Michigan State University, East Lansing, Michigan 48824, USA

(Received 29 May 2007; accepted 10 July 2007; published online 18 September 2007)

This paper presents a novel method for solving the Poisson-Boltzmann (PB) equation based on a rigorous treatment of geometric singularities of the dielectric interface and a Green's function formulation of charge singularities. Geometric singularities, such as cusps and self-intersecting surfaces, in the dielectric interfaces are bottleneck in developing highly accurate PB solvers. Based on an advanced mathematical technique, the matched interface and boundary (MIB) method, we have recently developed a PB solver by rigorously enforcing the flux continuity conditions at the solvent-molecule interface where geometric singularities may occur. The resulting PB solver, denoted as MIBPB-II, is able to deliver second order accuracy for the molecular surfaces of proteins. However, when the mesh size approaches half of the van der Waals radius, the MIBPB-II cannot maintain its accuracy because the grid points that carry the interface information overlap with those that carry distributed singular charges. In the present Green's function formalism, the charge singularities are transformed into interface flux jump conditions, which are treated on an equal footing as the geometric singularities in our MIB framework. The resulting method, denoted as MIBPB-III, is able to provide highly accurate electrostatic potentials at a mesh as coarse as 1.2 Å for proteins. Consequently, at a given level of accuracy, the MIBPB-III is about three times faster than the APBS, a recent multigrid PB solver. The MIBPB-III has been extensively validated by using analytically solvable problems, molecular surfaces of polyatomic systems, and 24 proteins. It provides reliable benchmark numerical solutions for the PB equation. © *2007 American Institute of Physics*. [DOI: 10.1063/1.2768064]

I. INTRODUCTION

Electrostatic interactions are omnipresent in biomolecules.[1,2] About 45% of aminoacids in globulare proteins are either ionized under physiological conditions or with polar groups in their side chains.[3] A high electron charge density of one charge per 1.7 Å is found in double-strained DNA chains. Electrostatics plays a paramount role in the structure, function, stability, and dynamics of macromolecules, such as signal transduction and other metabolic processes. The interaction of each pair of (partially) charged particles is governed by the Coulomb potential, which has a nonvanishing impact over a wide range of distance. Therefore, accurate and efficient evaluation of electrostatics has been one of the most challenging issues in computational molecular biology. Although a variety of methods, including Ewald summations, Euler summations, periodic images, and reaction field theory, have been developed in the past few decades, explicit electrostatic calculations of biomolecules in the solvent remain extremely expensive. Alternative implicit solvent models[4,5] have become very popular[6–12] since the pioneering work by Warwicker and Watson in the early 1980s,[13] and Honig and Nicholls in the 1990s.[8] The implicit solvent theory retains a microscopic treatment of biomolecules, while adopts a macroscopic mean-field description of the solvent. In such an approach, the Poisson-Boltzmann equation (PBE), or Poisson equation (PE) if no salt is present, is solved for electrostatic potentials. Consequently, electrostatic free energy of solvation,[14–16] pK_α values,[17–19] and electrostatic forces for molecular dynamics[20,21] can be calculated.

Although the PBE can be analytically solved for a few simple cases, it relies on numerical approaches to obtain useful solutions for realistic biological systems. A vast variety of computational approaches, such as finite difference methods,[7,8,10,13,20,22–24] finite element methods,[6,25–27] and boundary integral methods,[11,12,28–33] have been developed in the past few decades. Among these approaches, Cartesian grid based finite difference methods are commonly used in many popular software packages, such as DELPHI,[34,35] UHBD,[36] MEAD,[35]. APBS,[27,37,38] and CHARMM.[39] It was argued by Baker that finite difference based PB solvers, particularly in conjunction with multigrid linear algebraic solvers, can offer the best combination of speed, accuracy, and efficiency, making them the most popular approaches in structural biology.[4]

Implicit solvent models require a solvent-molecule interface to separate the solvent domain from the biomolecular domain. The molecular surface[40,41] is commonly used in the PBE for this purpose. The dielectric constants of the solvent domain and molecular domain are usually chosen as 80 and 1 (or 2), respectively, leading to discontinuous coefficients in the PBE. Moreover, molecular surfaces admit geometric singularities,[40,42–44] such as cusps and self-intersecting surfaces. Explicit interface treatment of geometric singularities

[a)]Author to whom correspondence should be addressed. Also at the Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824. Electronic mail: wei@math.msu.edu

has not been considered in the literature. Consequently, no Poisson-Boltzmann (PB) solver of second order convergence, i.e., the numerical error reduces by a factor of 4 when the grid spacing is halved, has ever been reported in the context of molecular surfaces of macromolecules. Indeed, irregular interfaces and geometric singularities cause numerical instability and slow down the convergence of existing PB solvers. Therefore, enhancing the stability and accelerating the convergence of PB solvers are pressing issues in developing the next generation PB solvers.

The rigorous treatment of discontinuous coefficients and singular sources in elliptic equations is a challenging task in applied mathematics and scientific computing. Since Peskin's pioneer work on immersed boundary method,[45] a number of other elegant methods have been constructed. Among these approaches, the ghost fluid method proposed by Fedkiw *et al.*[46] and Liu *et al.*[47] is relatively simple and easy to use for complex geometry. Finite element formulations are more suitable for weak solutions.[48–51] An upwinding embedded boundary method was proposed by Cai and Deng[52] for electromagnetic waves in dielectric media. Gibou and Fedkiw[53] proposed a fourth order extrapolation scheme for solving Laplace and heat equations on irregular domains. An integral equation approach was developed for complex geometries.[54] LeVeque and Li[55] proposed a remarkable second order sharp interface scheme, the immersed interface method. Their method has been applied to many practical problems, including the two-dimensional (2D) PBE.[56] Recently, we have proposed a matched interface and boundary (MIB) method for interface problems.[57–59] The MIB is of arbitrarily high order in principle. Fourth and sixth order MIB schemes have been demonstrated for curved interfaces.[58,59] This technique was used to developed our first generation MIB based PB solver, denoted as MIBPB-I, for arbitrary biomolecular interfaces.[60] Nevertheless, none of the aforementioned interface techniques is able to maintain its designed order of convergence and accuracy in the presence of geometric singularities. Indeed, technically, constructing higher-order convergence schemes for elliptic equations with geometric singularities is extremely challenging, despite great desire for doing so in practical applications. To our knowledge, the best result in the literature is of 0.8th order convergence reported by Hou and Liu,[61] achieved with a finite element formulation in 2D. More recently, we have extended the MIB method for solving elliptic equations with geometric singularities in 2D.[62] Most recently, we have constructed a second order MIB scheme for solving the PBE with geometric singularities of molecular surfaces.[63] Extensive numerical experiments indicate that our MIB based PB solver, denoted as MIBPB-II, outperforms traditional PB solvers in terms of accuracy and efficiency.[63]

A remaining issue in our MIB based PB solver is the efficient treatment of charge singularities. Technically, in most finite difference based PB solvers, Dirac delta functions in the PBE, or singular charges in a biomolecule, are redistributed to their neighboring grid points. This treatment works well in commonly used PB solvers where the major error is produced due to the neglect of interface continuity conditions. However, it affects the numerical accuracy of our MIBPB-II solver because of the possible interference of geometric interface and charge singularities. Stated differently, a set of auxiliary grid points that carry interface jump conditions may also carry singular charges, which leads to an accuracy reduction. This typically happens when the mesh size approaches half of the van der Waals radius. Consequently, at a given level of accuracy, our MIBPB-II delivers highly accurate electrostatic potentials when the grid spacing is smaller than 0.6 Å, but it behaves like a normal PB solver when a coarser mesh is used.

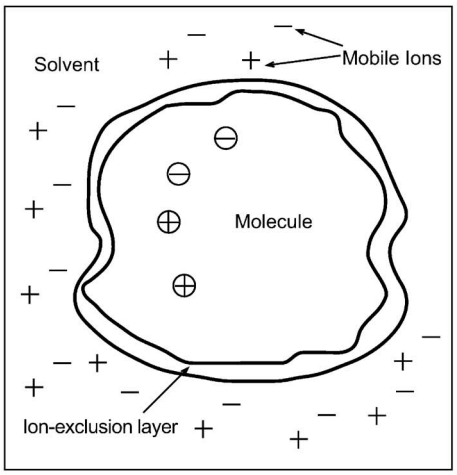
The objective of the present work is to overcome this obstacle by introducing Green's function technique into our MIB formulation. The separation of the PB solution into regular and singular parts was suggested by Zhou *et al.*[64] The feedback of the singular part to the regular part was not included in their formulation. Chern *et al.* have formulated the problem in a more rigorous manner.[65] However, they have only considered 2D problems with smooth interfaces. In the present work, we consider a Green's function formulation for the molecular surfaces of proteins with possible geometric singularities. The Green's function approach effectively transfers the contribution of charge singularities into a set of interface jump conditions which are determined by solving the corresponding Laplace equation with given singular sources. The charge-induced interface jump conditions are perfectly compatible with our MIB techniques for complex interfaces.[63,66] Therefore, we are able to treat geometric and charge singularities on an equal footing. The resulting PB solver, denoted as MIBPB-III, is capable of eliminating the interference of the geometric and charge singularities in solving the PBE and is of second order convergence. The present MIBPB-III is able to deliver high numerical accuracy at a grid spacing as large as the van der Waals radius (about 1.2 Å) for macromolecules. Consequently, it is about three times faster than APBS,[27,37,38] a recent multigrid PB solver.

The rest of the paper is organized as follows. Section II is devoted to the theoretical formulation and computational algorithm. The Green's function formulation of charge singularities is developed in the framework of our MIB method. A second order MIB scheme is constructed to solve the charge-induced boundary value problem with molecular surface singularities. The resulting MIBPB-III is extensively validated in Sec. III. The benchmark tests, such as the Kirkwood sphere,[67] the molecular surfaces of few-body systems, and 24 proteins, are used to examine the accuracy and test the speed of convergence of the proposed MIBPB-III. Comparisons are given to our previous MIBPB-II and two other established PB solvers. This article ends with a brief conclusion summarizing the main points.

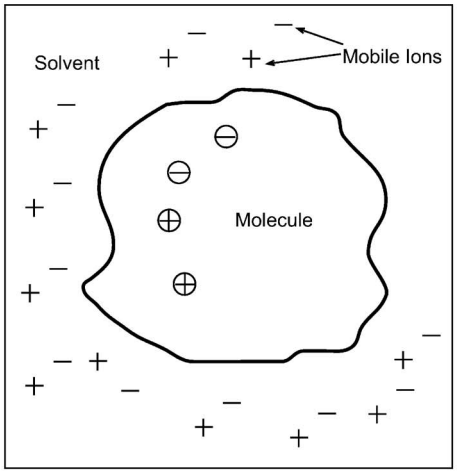
II. THEORY AND ALGORITHM

A. The Poisson-Boltzmann equation

Consider an open domain $\Omega \in \mathbb{R}^3$. Let Γ_{12} and Γ_{23} be two disjoint interfaces which divide Ω into three disjoint open subdomains, $\Omega=\Omega^- \cup \Omega^0 \cup \Omega^+$, see Fig. 1(a). Here Ω^- is the biomolecular domain, Ω^0 is a thin ion-exclusion layer, and Ω^+ is the solvent domain. By assuming the Boltzmann distribution for the equilibrium ionic electrostatic potential



(a)



(b)

FIG. 1. Domains of PBE. (a) three-domain model and (b) two-domain model.

energy $e_c\phi(\mathbf{r})$ in the dielectric continuum treatment for the ions in the solvent domain, the PBE takes the form[68]

$$-\nabla\cdot(\epsilon(\mathbf{r})\nabla\phi(\mathbf{r}))+\kappa^2(\mathbf{r})\left(\frac{k_BT}{e_c}\right)\sinh\left(\frac{e_c\phi(\mathbf{r})}{k_BT}\right)=4\pi\sum_{i=1}^{N_m}q_i\delta(\mathbf{r}-\mathbf{r}_i), \tag{1}$$

where $\phi(\mathbf{r})$ is the electrostatic potential and q_i is the (fractional) charge at position $\mathbf{r}_i$. Here, constants e_c, k_B, T, and N_m are, respectively, the electronic charge, Boltzmann constant, the absolute temperature, and number of charges in the biomolecule. The dielectric coefficient $\epsilon(\mathbf{r})$ and the ionic strength $\kappa(\mathbf{r})$, as functions of the space, are defined as

$$\epsilon(\mathbf{r})=\begin{cases}\epsilon^- & \mathbf{r}\in\Omega^-\\ \epsilon^+ & \mathbf{r}\in\Omega^0\cup\Omega^+\end{cases} \tag{2}$$

and

$$\kappa(\mathbf{r})=\begin{cases}0 & \mathbf{r}\in\Omega^-\cup\Omega^0\\ \bar{\kappa} & \mathbf{r}\in\Omega^+.\end{cases} \tag{3}$$

Equation (1) is subject to the following far-field boundary condition:

$$\phi(\infty)=0. \tag{4}$$

However, in practical computations, the Direchlet boundary condition

$$\phi(\mathbf{r})=\sum_{i=1}^{N_m}\frac{q_i}{\epsilon^+|\mathbf{r}-\mathbf{r}_i|}e^{-\bar{\kappa}|\mathbf{r}-\mathbf{r}_i|} \tag{5}$$

is used for the linearized PBE. Due to the discontinuous nature of coefficients ϵ and κ, Eq. (1) should be solved with following additional interface jump conditions across Γ_{12},

$$[\phi]_{\Gamma_{12}}=0, \tag{6}$$

$$[\epsilon\phi_{\mathbf{n}}]_{\Gamma_{12}}=0 \tag{7}$$

and across Γ_{23}

$$[\phi]_{\Gamma_{23}}=0, \tag{8}$$

$$[\phi_{\mathbf{n}}]_{\Gamma_{23}}=0, \tag{9}$$

where $\mathbf{n}$ is the outer normal direction of the interface. Mathematically, these jump conditions are required to ensure the uniqueness of the solution. Physically, these conditions are required by the continuity of the electrostatic potential and its flux across the interface.

Since the ion-exclusion layer is typically about 1 Å in thickness, the thin layer domain Ω^0 normally has very limited impact to the electrostatic potential. For simplicity, we adopt a two-domain model as illustrated in Fig. 1(b), where the entire domain is divided by the boundary of the molecule, Γ, into two disjoint domains Ω^- and Ω^+. Correspondingly, we consider the simplified $\epsilon(\mathbf{r})$ and $\kappa(\mathbf{r})$ as

$$\epsilon(\mathbf{r})=\begin{cases}\epsilon^- & \mathbf{r}\in\Omega^-\\ \epsilon^+ & \mathbf{r}\in\Omega^+\end{cases} \tag{10}$$

and

$$\kappa(\mathbf{r})=\begin{cases}0 & \mathbf{r}\in\Omega^-\\ \bar{\kappa} & \mathbf{r}\in\Omega^+.\end{cases} \tag{11}$$

The interface jump conditions across Γ are given as

$$[\phi]_{\Gamma}=0, \tag{12}$$

$$[\epsilon\phi_{\mathbf{n}}]_{\Gamma}=0. \tag{13}$$

B. Green's function formulation of the Poisson-Boltzmann equation

Numerically, second and higher-order numerical implementation of Dirac delta functions on Cartesian grid points is feasible with appropriate interpolation schemes. However, the overlap of grid points carrying redistributed charges and those involved in the treatment of geometric interface singularities leads to an accuracy reduction. This interference be-

comes inevitable when a coarse mesh is pursued in an interface treatment. Therefore, a Green's function approach of the singular charges becomes attractive in our higher-order interface schemes for the linearized PBE. Such an approach decomposes the solution ϕ into regular part $\tilde{\phi}$ and singular part $\bar{\phi}$.[64] The latter consists of a fundamental solution of the Poisson equation with the singular charge, ϕ^*, and a harmonic function ϕ^0,[65]

$$\phi = \tilde{\phi} + \bar{\phi}, \tag{14}$$

where $\bar{\phi}(\mathbf{r})$ is defined as

$$\bar{\phi}(\mathbf{r}) = \begin{cases} \phi^*(\mathbf{r}) + \phi^0(\mathbf{r}) & \mathbf{r} \in \Omega^- \\ 0 & \mathbf{r} \in \Omega^+. \end{cases} \tag{15}$$

Here $\phi^*(\mathbf{r})$ is the Green's function

$$\phi^*(\mathbf{r}) = \sum_{i=1}^{N_m} \frac{q_i}{\epsilon^- |\mathbf{r} - \mathbf{r}_i|} \tag{16}$$

solving the Poisson equation with singular charges, while $\phi^0(\mathbf{r})$ is a harmonic function in Ω^- satisfying the Laplace equation

$$\begin{cases} \nabla^2 \phi^0(\mathbf{r}) = 0 & \text{in } \Omega^- \\ \phi^0(\mathbf{r}) = -\phi^*(\mathbf{r}) & \text{on } \Gamma. \end{cases} \tag{17}$$

The boundary value problem given by Eq. (17) is to be solved with designed order of convergence over the irregular domain Ω^- with possibly geometric singularities. As a solution to the PBE, the singular part, $\bar{\phi}(\mathbf{r})$, is solved subject to the following jump conditions:

$$[\bar{\phi}]_\Gamma = 0 \quad \text{and} \quad [\epsilon \overline{\phi_{\mathbf{n}}}]_\Gamma = -\epsilon^- \nabla (\phi^* + \phi^0) \cdot \mathbf{n}|_\Gamma. \tag{18}$$

We work on the linearized version of Eq. (1); thus the equation for the correction potential $\tilde{\phi}(\mathbf{r}) = \phi(\mathbf{r}) - \bar{\phi}(\mathbf{r})$ is homogeneous,

$$-\nabla \cdot (\epsilon(\mathbf{r}) \nabla \tilde{\phi}(\mathbf{r})) + \kappa^2(\mathbf{r})(\bar{\phi}(\mathbf{r}) + \tilde{\phi}(\mathbf{r})) = 0, \tag{19}$$

and can be rewritten as

$$-\nabla \cdot (\epsilon(\mathbf{r}) \nabla \tilde{\phi}(\mathbf{r})) + \kappa^2(\mathbf{r}) \tilde{\phi}(\mathbf{r}) = -\kappa^2(\mathbf{r}) \bar{\phi}(\mathbf{r}). \tag{20}$$

This equation is solved with jump conditions

$$[\tilde{\phi}]_\Gamma = 0 \quad \text{and} \quad [\epsilon \tilde{\phi}_{\mathbf{n}}]_\Gamma = -[\epsilon \bar{\phi}_{\mathbf{n}}]_\Gamma = \epsilon^- \nabla (\phi^* + \phi^0) \cdot \mathbf{n}|_\Gamma. \tag{21}$$

The interface problem given by Eq. (20) with interface jump conditions (21) and appropriate far-field boundary conditions is to be solved with designed order of convergence. The MIB method is generalized for this purpose.

C. The matched interface and boundary (MIB) method

To solve the elliptic interface problem given by Eq. (20), we use the MIB method. Let us consider an elliptic equation of the form

$$-\nabla \cdot (\epsilon(\mathbf{r}) \nabla \phi(\mathbf{r})) + \kappa^2(\mathbf{r}) \phi(\mathbf{r}) = f(\mathbf{r}). \tag{22}$$

The interface Γ divides the whole domain into two separated parts, Ω^- and Ω^+. The jump conditions across the interface are given by

$$[\phi] = \phi^+(\mathbf{r}) - \phi^-(\mathbf{r}), \tag{23}$$

$$[\epsilon \phi_{\mathbf{n}}] = \epsilon^+(\mathbf{r}) \nabla \phi^+(\mathbf{r}) \cdot \mathbf{n} - \epsilon^-(\mathbf{r}) \nabla \phi^-(\mathbf{r}) \cdot \mathbf{n}, \tag{24}$$

where $\mathbf{n} = (n_x, n_y, n_z)$ is the outer normal direction of the interface Γ. Here $[\phi]$ and $[\epsilon \phi_{\mathbf{n}}]$ are given or can be computed from other quantities. Appropriate far-field boundary conditions are also prescribed for Eq. (22) and the outer boundary of Ω^+ is chosen to be regular. In our MIB method, a standard Cartesian grid is used along with the standard finite difference schemes. However, near the interface, the standard finite difference schemes lose their designed convergence. We therefore implement the interface jump conditions to restore the accuracy. To this end, we classify all the grid points into two classes, the regular ones and the irregular ones. An irregular grid point is defined as one where the standard finite difference scheme involves grid points across the interface. Obviously, for a given grid point near the interface, it may and may not be an irregular point, depending on the order of the finite difference scheme used. For a given finite difference scheme, all the irregular grid points on one side of the interface constitute a fictitious domain, see Fig. 2. Fictitious values, which are the linear combination of regular points and jump conditions, at irregular points are computed as a continuation of the solution from the other side of the interface by using the jump conditions. The fictitious value is used when a finite difference scheme reaches across the interface. In the following, we discuss how to calculate fictitious values.

Assume that the interface Γ intersects the grid in the x direction at a point (i_o, j, k), which is located between (i, j, k) and $(i+1, j, k)$. We therefore have two irregular grid points, (i, j, k) and $(i+1, j, k)$, for the second order finite difference scheme in the x direction near the interface. The fictitious values at these two irregular points, $f^+(i, j, k)$ and $f^-(i+1, j, k)$, are to be determined. However, one of the jump conditions, Eq. (24), is defined in the normal direction of the interface at point (i_o, j, k). It is convenient to introduce a local coordinates (ξ, η, ζ) such that ξ is along the normal direction and η is in the x-y plane. The coordinate transformation can be given as

$$\begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} = \mathbf{p} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \tag{25}$$

where $\mathbf{p}$ is the transformation matrix

$$\mathbf{p} = \begin{bmatrix} \sin\psi\cos\theta & \sin\psi\sin\theta & \cos\psi \\ -\sin\theta & \cos\theta & 0 \\ -\cos\psi\cos\theta & -\cos\psi\sin\theta & \sin\psi \end{bmatrix}. \tag{26}$$

Here θ and ψ are the azimuth and zenith angles with respect to the normal direction ξ, respectively. By differentiating

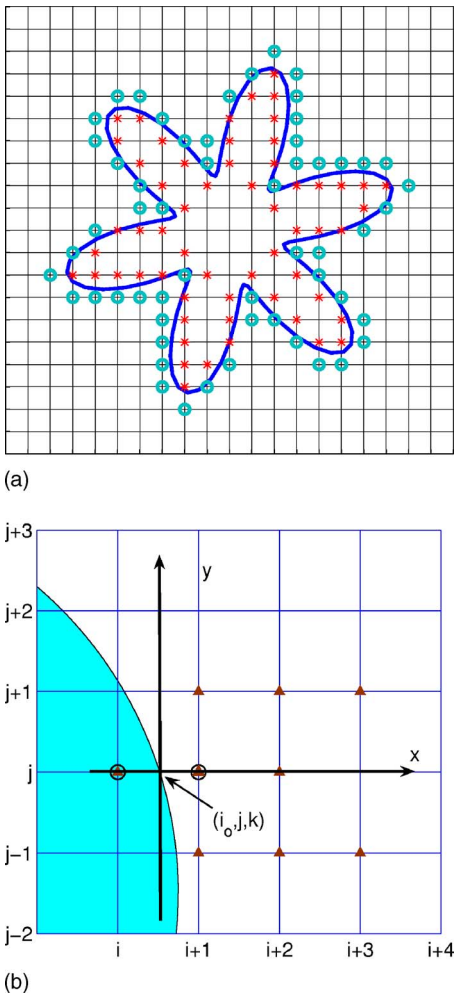


FIG. 2. (Color online) An illustration of the MIB method: (a) fictitious domains constructed with irregular grid points in a second order MIB scheme and (b) MIB scheme.

Eq. (23) along two tangential directions, η and ζ, we can generate two additional jump conditions

$$[\phi_\eta] = (\phi_x^+ p_{21} + \phi_y^+ p_{22} + \phi_z^+ p_{23}) - (\phi_x^- p_{21} + \phi_y^- p_{22} + \phi_z^- p_{23}), \tag{27}$$

$$[\phi_\zeta] = (\phi_x^+ p_{31} + \phi_y^+ p_{32} + \phi_z^+ p_{33}) - (\phi_x^- p_{31} + \phi_y^- p_{32} + \phi_z^- p_{33}), \tag{28}$$

where p_{ij} is the ijth component of the transformation matrix $\mathbf{p}$. Since Eq. (25) implies

$$\begin{bmatrix} \phi_\xi \\ \phi_\eta \\ \phi_\zeta \end{bmatrix} = \mathbf{p} \cdot \begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix}, \tag{29}$$

Eqs. (24), (27), and (28) can be rewritten as follows:

$$\begin{bmatrix} [\beta\phi_\xi] \\ [\phi_\eta] \\ [\phi_\zeta] \end{bmatrix} = \mathbf{C} \cdot \begin{bmatrix} \phi_x^+ \\ \phi_x^- \\ \phi_y^+ \\ \phi_y^- \\ \phi_z^+ \\ \phi_z^- \end{bmatrix}, \tag{30}$$

where

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

where $\mathbf{C}_i$ represents the ith row of matrix $\mathbf{C}$. It is clear that none of these three jump conditions is easy to implement because there are six derivatives, ϕ_x^+, ϕ_x^-, ϕ_y^+, ϕ_y^-, ϕ_z^+, and ϕ_z^-, to be calculated in appropriate subdomains near the interface. For complex solvent-molecule interfaces of macromolecules, it is often very difficult to evaluate some of these derivatives. Therefore, in the MIB method, we avoid calculating two most difficult derivatives by eliminating them with two relevant jump conditions. Figure 2(b) illustrates 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 $(\phi_x^+, \phi_x^-, \phi_y^+, \phi_y^-, \phi_z^+, \phi_z^-)$, Eq. (30) becomes

$$a[\beta\phi_\xi] + b[\phi_\eta] + c[\phi_\zeta] = (a\mathbf{C}_1 + b\mathbf{C}_2 + c\mathbf{C}_3) \cdot \begin{bmatrix} \phi_x^+ \\ \phi_x^- \\ \phi_y^+ \\ \phi_y^- \\ \phi_z^+ \\ \phi_z^- \end{bmatrix}, \tag{32}$$

where

$$\begin{aligned} a &= C_{2l}C_{3m} - C_{3l}C_{2m}, \\ b &= C_{3l}C_{1m} - C_{1l}C_{3m}, \\ c &= C_{1l}C_{2m} - C_{2l}C_{1m}. \end{aligned} \tag{33}$$

We therefore use Eqs. (23) and (32) to determine two fictitious values near the interface along a specific mesh line at a time. This procedure is systematically repeated to determine fictitious values along other mesh lines and at other interface locations. In this manner, we have effectively reduced a three-dimensional (3D) interface problem into a one-dimensional (1D)-like one.

Returning to the situation in Fig. 2(b), after eliminating ϕ_y^-, ϕ_y^+ is evaluated by using interpolation schemes with solution values from the neighboring points. In detail, ϕ_y^+ at (i_o, j, k) will be interpolated by using grid value at points

$(i_o,j-1,k)$, (i_o,j,k), and $(i_o,j+1,k)$. Since these three points are not normal grid points, we need to further interpolate the values at $(i_o,j-1,k)$ by using grid values at $(i+1,j-1,k)$, $(i+2,j-1,k)$, and $(i+3,j-1,k)$, 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 applied for the interpolation. 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. In fact, care is required to select appropriate interpolation schemes so that the resulting MIB matrix is optimally symmetric and diagonally dominant for arbitrarily complex solvent-molecule interfaces with geometric singularities. A detailed discussion of the matrix optimization and 3D MIB methods for geometric singularities is given in Ref. 66. A rigorous validation of MIB based PB solver, MIBPB-II, for molecular surface singularities can be found in Ref. 63.

D. Solution to the boundary value problem

To obtain ϕ^0, we need to solve the boundary value problem given by Eq. (17). We use the standard seven-point finite difference scheme on all the regular grid points. However, for irregular grid points in Ω^-, the standard seven-point finite difference scheme requires fictitious value(s) outside the interface. In this case, we could determine the fictitious values by extrapolation involving the interface values. However, instead of using the fictitious values outside the interface, we use the interface values at the intersecting points of the mesh and the interface. Since the resulting finite difference weights of the seven-point scheme are not strictly symmetric, this treatment slightly reduces the accuracy of the scheme at the grid points close to the interface. However, we have tested that this local accuracy reduction does not affect the overall second order convergence of our scheme. Moreover, the resulting matrix is nearly symmetric and diagonally dominant, and can be efficiently solved by a preconditioned biconjugate gradient solver.

E. The evaluation of jump conditions

To solve Eq. (20) with the MIB method, we need to evaluate the jump conditions in Eq. (21). In other words, after solving the boundary value problem (17), we need to calculate the derivatives of ϕ^0 and ϕ^* in the normal direction at each intersecting point of the mesh lines and the interface. According to Eq. (30), these normal derivatives can be obtained from the partial derivatives of ϕ^0 and ϕ^* in the x, y, and z directions by means of the transformation matrix. We encounter the difficulties that both ϕ^0 and ϕ^* are available only inside the interface Γ. In fact, the Green's function ϕ^* is analytically available and its gradients can be computed to arbitrarily high accuracy at each intersecting point. Therefore, the accuracy is often limited by approximating the derivatives of ϕ^0. We seek to extend domain Ω^- by extrapolations so that central difference schemes can be applied, since central difference schemes are more accurate than the asymmetric ones for a given length of stencils. Moreover, in many cases with geometric singularities and restricted geometric environment, a domain extension becomes inevitable. Derivatives are computed on the enlarged domain. Various techniques developed in our MIB are utilized to calculate these derivatives. This procedure is discussed as follows.

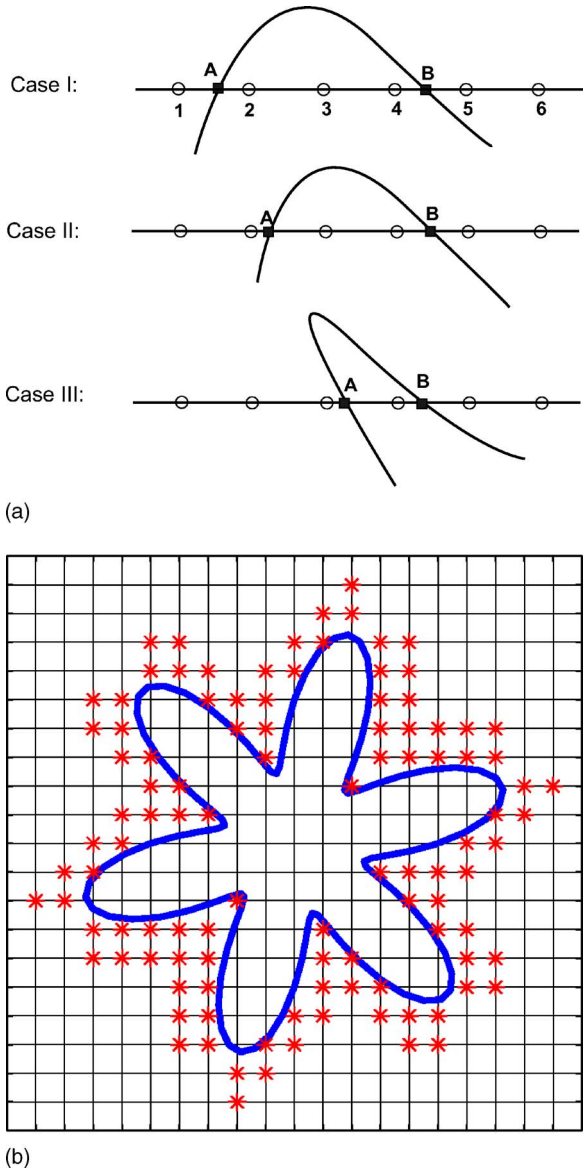


FIG. 3. (Color online) (a) ϕ^0 extension schemes and (b) illustration.

1. The extension of ϕ^0 in the fictitious domain

Having solved the boundary value problem (17), ϕ^0 is available in Ω^-. To evaluate partial derivatives of ϕ^0 at each intersecting point of the mesh lines and the interface, we extend domain Ω^- by extrapolation. Figure 3(a) gives three cases of a grid point close to the interface and the schemes used to extend the domain. In Case I, there are at least three grid points inside the interface in the direction of extension. We just use three ϕ^0 on grid points (2,3,4) and the value on the interface point B to extrapolate the function value on grid point 5, and then use ϕ^0 on points $(2,3,4,B)$ and the extrapolated function on grid point 5 to extrapolate the function value on grid point 6. In Case II, there are only two grid

points in between the interfaces in the direction of extension. We use ϕ^0 on points $(A,3,4,B)$ to extrapolate ϕ^0 on grid point 5, and then use ϕ^0 on points $(A,3,4,B)$ plus the extrapolated ϕ^0 on grid point 5 to extrapolate ϕ^0 on grid point 6. In Case III, there is only one grid point in between the interfaces in the direction of extension. We will use ϕ^0 on points $(A,4,B)$ to extrapolate ϕ^0 on grid point 5. Due to the fact we only have three points here and they are nonuniformly distributed, we will only extend ϕ^0 to one grid point to maintain the accuracy of the extension.

There is a choice to be made when ϕ^0 on two extended grid points can be obtained from extensions of different directions. It is found that the accuracy of the extension is optimized if the following order of priorities is maintained: Case I for the first point, Case II for the first point, Case III for the first point, Case I for the second point, and Case II for the second point. We have examined the accuracy of this order by using many test calculations with different geometric variations of the interfaces. In a smooth geometry with a sufficiently dense grid, Case I will be dominant and Case III rarely occurs. Although our extensions involve schemes of different convergence orders, these approaches are optimal or nearly optimal for geometric singularities, which are extremely challenging. The overall scheme is of numerically second order even for complex molecular surfaces of twenty four proteins.

After the extension, we obtain a fictitious domain outside the interface as illustrated in Fig. 3(b). The computations of partial derivatives ϕ_x^0, ϕ_y^0, and ϕ_z^0 are carried out.

2. The calculation of partial derivatives

To calculate the derivative of ϕ^0 in the normal direction, we need to compute three partial derivatives, ϕ_x^0, ϕ_y^0, and ϕ_z^0, at each intersecting point of the mesh lines and the interface. Since central schemes or nearly central schemes are preferred, these derivatives are calculated by using ϕ^0 in the domain Ω^0, on the interface Γ, and in the extended fictitious domain. Figure 4 gives an x-y cross section illustration of the schemes for computing $\phi_x^0(i_o,j,k)$ and $\phi_y^0(i_o,j,k)$ at the intersecting point of the jth mesh line and the interface and near the grid point (i,j,k). In Fig. 4(a), the area circled by the solid curve Γ is the domain Ω^-, and the area between the solid curve and the dashed curve is the extended domain. If we restrict ourselves to grid points inside the domain Ω^-, we can compute ϕ_x^0 by $\phi^0(i-2,j,k)$, $\phi^0(i-1,j,k)$, and $\phi^0(i,j,k)$ but it will be impossible for us to calculate $\phi_y^0(i_o,j,k)$ directly. We therefore have to consider ϕ^0 in the extended domain and on the interface. In this scheme, ϕ_x^0 can be evaluated by using $\phi^0(i-1,j,k)$, $\phi^0(i,j,k)$, and $\phi^0(i+1,j,k)$. Similarly, $\phi_y^0(i_o,j,k)$ can be calculated by using $\phi^0(i_o,j-1,k)$, $\phi^0(i_o,j,k)$, and $\phi^0(i_o,j+1,k)$ as denoted by squares. However, $\phi^0(i_o,j-1,k)$ and $\phi^0(i_o,j+1,k)$ are not available and they are to be computed by interpolations with $(\phi^0(i-2,j-1,k)$, $\phi^0(i-1,j-1,k)$, $\phi^0(i,j-1,k))$ and $(\phi^0(i-1,j+1,k)$, $\phi^0(i,j+1,k)$, $\phi^0(i+1,j+1,k))$, respectively.

Very rarely but possibly, we may encounter more challenging situations as illustrated in Fig. 4(b). This typically occurs with geometric singularities. In this case, $\phi_y^0(i_o,j,k)$ cannot be computed directly, nor can it be computed with $\phi^0(i_o,j-1,k)$, $\phi^0(i_o,j,k)$, and $\phi^0(i_o,j+1,k)$ because $\phi^0(i_o,j-1,k)$ cannot be computed even with the extended fictitious domain. To overcome this difficulty, we propose an alternative scheme. We first compute $\phi_y^0(i,j,k)$ by using $\phi^0(i,j-1,k)$, $\phi^0(i,j,k)$, and $\phi^0(i,j+1,k)$. We then calculate $\phi_y^0(i+1,j,k)$ by using $\phi^0(i+1,j-1,k)$, $\phi^0(i+1,j,k)$, and $\phi^0(i+1,j+1,k)$. Finally, we use $\phi^0(i+2,j,k)$, $\phi^0(i+2,j+1,k)$, and $\phi^0(i+2,j+2,k)$ to compute $\phi_y^0(i+2,j,k)$. Eventually, $\phi_y^0(i_o,j,k)$ can be obtained by interpolations with $\phi_y^0(i,j,k)$, $\phi_y^0(i+1,j,k)$, and $\phi_y^0(i+2,j,k)$.

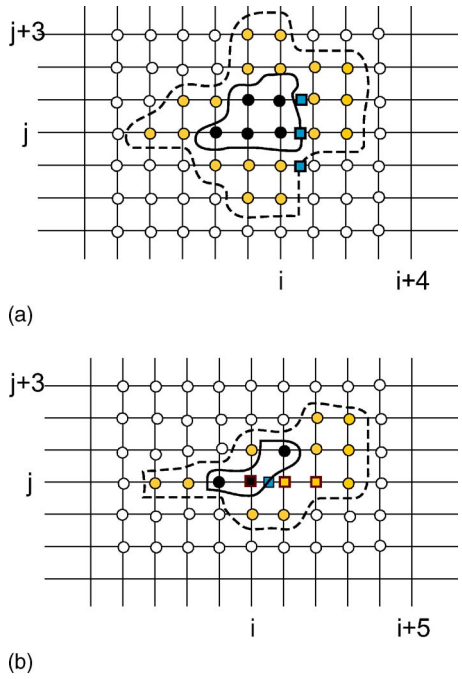


FIG. 4. (Color online) Schemes to find partial derivatives: (a) regular cases and (b) a special case.

F. The computation of electrostatic free energy of solvation

The calculation of the electrostatic free energy of solvation plays a critical role in the study of biomolecule in solvent. Such calculation is also one of the most important applications of the PBE. In the Green's function formulation, the evaluation of the electrostatic free energy of solvation is slightly different from that in the direct solution of the PBE

$$\Delta G_{\mathrm{sol}} = \frac{1}{2}\sum_{i=1}^{N_m} q(\mathbf{r}_i)(\phi_{\mathrm{diele}}(\mathbf{r}_i) - \phi_{\mathrm{homo}}(\mathbf{r}_i)), \tag{34}$$

where $q(\mathbf{r}_i)$ is the partial charges at $\mathbf{r}_i \in \Omega^-$, and ϕ_{diele} and ϕ_{homo} are electrostatic potentials in the dielectric and homogeneous environments, respectively. In MIBPB-II and many other finite difference based methods, the singular charges are redistributed to the neighboring grid points and the summation is computed over all of these redistributed charges. Electrostatic potential in homogeneous environment ϕ_{homo} is obtained by the fast Fourier transform (FFT).

TABLE I. Comparison of electrostatic free energies and surface potential errors for a unit charge in a dielectric sphere (Case 1).

	PBEQ			APBS			MIBPB-II			MIBPB-III		
h	ΔG	E_1	E_2	ΔG	E_1	E_2	ΔG	E_1	E_2	ΔG	E_1	E_2
1.00	−83.57	25.23	91.07	−83.44	25.31	90.87	−83.68	7.31	52.37	−81.95	1.62E−1	4.890
0.50	−85.78	17.06	84.26	−85.85	17.05	84.26	−81.97	2.56	6.68	−81.98	2.49E−2	0.966
0.20	−82.84	7.50	74.44	−82.58	7.51	74.43	−81.98	3.01E−1	1.57	−81.98	5.05E−3	0.161
0.10	−82.49	3.83	63.20	−82.27	3.84	63.20	−81.98	3.89E−2	0.37	−81.98	1.39E−3	0.050
0.05	−82.20	1.89	46.95	−82.03	1.94	46.18	−81.98	9.96E−3	0.086	−81.98	3.02E−4	0.013

In the Green's function formulation, the electronic potential in the homogeneous environment ϕ_{homo} can be identified with the Green's function which solves the Poisson equation with the singular charge source term, ϕ^*. Since ϕ^* is also a part of the full solution as shown in Eq. (14), it does not contribute to the electrostatic free energy of solvation. The latter, therefore, can be directly computed from the sum of $\tilde{\phi}$ and ϕ^0,

$$\Delta G_{\text{sol}} = \frac{1}{2}\sum_{i=1}^{N_m} q(\mathbf{r}_i)(\tilde{\phi}(\mathbf{r}_i) + \phi^0(\mathbf{r}_i)). \tag{35}$$

Note that ϕ^0 is defined in a smaller domain Ω^- and its evaluation is relatively cheap.

III. RESULTS AND DISCUSSIONS

In this section, we examine the accuracy, validate the convergence, and test the speed of the proposed MIBPB-III solver. For a comparison, we also applied some other established methods, such as PBEQ,[24] a representative finite difference PB solver from CHARMM,[39] and APBS,[27,37,38] a recently developed multigrid finite element and finite difference PB solver. The finite difference function of the APBS is utilized in our calculations. The comparison is also given to our second generation MIB based PB solver, the MIBPB-II, which provides a rigorous treatment of geometric singularities. The specific MIBPB-II solver used in this comparison is a new code. Extensive experiments are carried out over Kirkwood's dielectric sphere,[67] molecular surfaces of 1, 2, and 18 atoms, and a set of 24 proteins used in previous tests.[60,69] Molecular surfaces are generated by using the MSMS (version 2.5.7) (Ref. 44) at density 10 and with a probe radius of 1.4 Å unless specified. In all test cases, the dielectric constants are taken as $\epsilon^+=80$ and $\epsilon^-=1$. The latter is the same as the dielectric constant in the vacuum, ϵ_0. For all protein structures hydrogen atoms were added to obtain full all-atom models. Partial charges at atomic sites and atomic van der Waals radii were taken from the CHARMM22 force field.[70]

To compare the computational performance, we use three error measurements, the surface maximum absolute error E_1, the surface maximum relative error E_2, and the surface root mean square error E_3:

$$E_1 = \max_\Gamma |\phi(x,y,z) - \hat{\phi}(x,y,z)|,$$

$$E_2 = 100\max_\Gamma \left| \frac{\phi(x,y,z) - \hat{\phi}(x,y,z)}{\hat{\phi}(x,y,z)} \right|,$$

$$E_3 = \sqrt{\frac{1}{N_{\text{irr}}}\sum_{i=1}^{N_{\text{irr}}} |\phi_i - \hat{\phi}_i|^2},$$

where ϕ and $\hat{\phi}$ are numerical and exact solutions, respectively. The maximum value and summation are taken over all the irregular grid points near the interface Γ, whose number is given by N_{irr}.

A. Validation

1. Kirkwood dielectric sphere

To validate the proposed MIBPB-III, we first consider Kirkwood's dielectric sphere of radius 2.0 Å. In our tests, the number of charges and their locations are allowed to vary over a wide range. A detailed derivation of the model is given in the Appendix.

Case 1: A dielectric sphere with a centered charge. We start our test with a single unit charge at the center of a sphere. For this case, ϕ^* will be a function of the radius r; therefore it will have the constant value $\phi^*=1/b\epsilon^-$ on the interface, where b is the radius of the spherical biomolecule. This constant value of ϕ^* on the interface will lead to a constant $\phi^0=-1/b\epsilon^-$ over the domain Ω^-. Since ϕ^0 is a constant, the interface jump condition (21) of Eq. (20) will depend only on the normal derivative of ϕ^*, which is analytically given. Table I lists the results of the potential near interface and the electrostatic free energies of solvation of MIBPB-III and two other established PB solvers, PBEQ and APBS. Results of our previous MIBPB-II are also given to show the consistency and improvement of the present MIBPB-III scheme. In this table, ΔG is the free energy of solvation with exact value of −81.98 kcal/mol. All the solvation energies are reported in kcal/mol. Meanwhile, all electrostatic potentials are reported in kcal/mol/e_c unless specified.

At the mesh size of $h=0.5$ Å, both MIBPB-II and MIBPB-III are able to provide very accurate electrostatic free energy of solvation. In fact, their errors at $h=0.5$ Å are smaller than those of APBS and PBEQ at the mesh size as small as $h=0.05$ Å. These results indicate the impact and importance of rigorous mathematical treatment of interface flux continuity conditions.

At the mesh size of $h=1.0$ Å, MIBPB-II loses its accuracy in the electrostatic free energy of solvation because it

TABLE II. Numerical errors of surface electrostatic potentials for an off-centered charge in a dielectric sphere (Case 2).

		PBEQ		MIBPB-II		MIBPB-III	
a	h	E_1	Order	E_1	Order	E_1	Order
0.2	1.0	2.96*E*+01		1.09*E*+01		1.93*E*−01	
	0.5	2.16*E*+01	0.46	7.33*E*+00	0.58	3.55*E*−02	2.44
	0.2	9.97*E*+00	0.84	4.50*E*−01	3.05	7.17*E*−03	1.75
	0.1	5.15*E*+00	0.95	6.00*E*−02	2.91	2.01*E*−03	1.84
0.4	1.0	3.35*E*+01		1.31*E*+01		2.23*E*−01	
	0.5	2.76*E*+01	0.28	6.13*E*+00	1.09	4.38*E*−02	2.35
	0.2	1.33*E*+01	0.80	7.10*E*−01	2.35	9.23*E*−03	1.70
	0.1	6.99*E*+00	0.93	1.00*E*−01	2.83	2.76*E*−03	1.74
0.6	1.0	3.59*E*+01		1.27*E*+01		2.19*E*−01	
	0.5	3.39*E*+01	0.08	7.17*E*+00	0.83	4.48*E*−02	2.29
	0.2	1.79*E*+01	0.70	1.17*E*+00	1.98	1.02*E*−02	1.61
	0.1	9.46*E*+00	0.92	1.60*E*−01	2.87	3.52*E*−03	1.54
0.8	1.0	3.62*E*+01		1.10*E*+01		2.44*E*−01	
	0.5	4.05*E*+01	−0.16	1.00*E*+01	0.14	1.29*E*−01	0.92
	0.2	2.43*E*+01	0.56	2.01*E*+00	1.75	1.91*E*−02	2.08
	0.1	1.33*E*+01	0.87	3.20*E*−01	2.65	3.79*E*−03	2.33
1.0	1.0	3.39*E*+01		2.51*E*+01		7.94*E*−01	
	0.5	4.61*E*+01	−0.44	1.85*E*+01	0.44	4.82*E*−01	0.72
	0.2	3.29*E*+01	0.37	3.55*E*+00	1.80	8.07*E*−02	1.95
	0.1	1.96*E*+01	0.75	5.80*E*−01	2.61	1.48*E*−02	2.44
1.2	1.0	2.91*E*+01		9.87*E*+00		1.93*E*+00	
	0.5	4.81*E*+01	−0.72	3.12*E*+01	−1.66	1.38*E*+00	0.48
	0.2	4.36*E*+01	0.11	5.82*E*+00	1.83	2.88*E*−01	1.72
	0.1	2.88*E*+01	0.60	1.34*E*+00	2.12	5.78*E*−02	2.32
1.5	1.0	1.86*E*+01		3.76*E*+00		6.41*E*+00	
	0.5	4.90*E*+01	−1.40	4.31*E*+01	−3.52	4.91*E*+00	0.38
	0.2	5.65*E*+01	−0.16	2.47*E*+01	0.61	1.95*E*+00	1.01
	0.1	5.02*E*+01	0.17	6.38*E*+00	1.95	5.35*E*−01	1.87

does not have a Green's function treatment of charge singularity. In this case, the grid points that carry the information of interface jump conditions are mixed with those that carry the distributed singular charges. Consequently, MIBPB-II has a similar level of errors as PBEQ and APBS, which do not have an explicit mathematical treatment of interface flux jump conditions. In contrast, the advantage of MIBPB-III is very obvious in this situation. Because of its Green's function treatment of singular charge and its rigorous treatment of interface flux jump conditions, the MIBPB-III is able to deliver high accuracy in the prediction of the electrostatic free energy of solvation at $h=1.0$ Å.

Surface electrostatic potentials are very important for protein-protein and protein-ligand interaction, and thus the electrostatic steering effects of signal transduction. It was not widely recognized that traditional PB solvers produce very large relative errors in the prediction of surface electrostatic potentials, as much as 70% at the mesh size of $h=0.2$ Å for a unit charge in a dielectric sphere. The earlier MIBPB-I and MIBPB-II are able to deliver highly accurate surface electrostatic potentials for this system, with less than 2% errors at $h=0.5$ Å. However at a coarse mesh size of $h=1.0$ Å, MIBPB-I and MIBPB-II cannot maintain their accuracy for the aforementioned reason. Our MIBPB-III is designed to overcome this difficulty. Indeed, MIBPB-III is able to provide extremely accurate surface electrostatic potentials at $h=1.0$ Å, as shown in Table I.

Case 2: A dielectric sphere with an off-centered charge. In Case 2, we allow the unit charge to locate at a distance a from the center of the sphere. We set mesh size h to 1.0, 0.5, 0.2, and 0.1 Å. Table II gives the numerical errors of surface potentials calculated by PBEQ, MIBPB-II, and MIBPB-III. The unit of the surface potentials is kcal/mol/e_c. From the table we can see that both MIBPB-II and MIBPB-III can approximately achieve second order accuracy. While MIBPB-III is more consistent in its convergence when the charge approaches the boundary of the sphere. When both a and h are relatively small, MIBPB-III is at least one order of magnitude more accurate than MIBPB-II. When $a=1.5$ and $h\geqslant 0.5$ Å, MIBPB-II is abnormal because the system is under resolved. MIBPB-III is still more accurate. In other cases, we see a pattern that MIBPB-III outperforms MIBPB-II, and MIBPB-II outperforms PBEQ.

Table III provides a detailed analysis of MIBPB-III er-

TABLE III. Component analysis of surface electrostatic potentials for an off-centered charge in a dielectric sphere (Case 2).

		Flux error		ϕ^0 error		$\tilde{\phi}$ error		ϕ error	
a	h	E_1	Order	L_∞	Order	L_∞	Order	E_1	Order
0.2	1.0	1.68E−04		3.05E−06		5.81E−04		5.81E−04	
	0.5	4.60E−05	1.87	7.13E−07	2.10	1.06E−04	2.45	1.07E−04	2.44
	0.2	1.07E−05	1.59	6.61E−08	2.60	2.16E−05	1.74	2.16E−05	1.75
	0.1	3.22E−06	1.73	9.21E−09	2.84	6.04E−06	1.84	6.04E−06	1.84
0.4	1.0	1.87E−03		4.32E−05		6.77E−04		6.72E−04	
	0.5	5.55E−04	1.75	1.45E−05	1.57	1.35E−04	2.33	1.32E−04	2.35
	0.2	9.87E−05	1.88	1.45E−06	2.51	2.79E−05	1.72	2.78E−05	1.70
	0.1	2.97E−05	1.73	2.12E−07	2.77	8.33E−06	1.74	8.32E−06	1.74
0.6	1.0	9.00E−03		2.51E−04		6.91E−04		6.60E−04	
	0.5	2.92E−03	1.62	9.59E−05	1.39	1.50E−04	2.20	1.35E−04	2.29
	0.2	5.34E−04	1.85	1.05E−05	2.41	3.13E−05	1.71	3.08E−05	1.61
	0.1	1.21E−04	2.14	1.56E−06	2.75	1.06E−05	1.56	1.06E−05	1.54
0.8	1.0	3.12E−02		8.96E−04		5.71E−04		7.34E−04	
	0.5	1.12E−02	1.48	4.04E−04	1.15	2.22E−04	1.36	3.88E−04	0.92
	0.2	2.10E−03	1.83	5.02E−05	2.28	3.78E−05	1.93	5.75E−05	2.08
	0.1	5.64E−04	1.90	7.82E−06	2.68	1.16E−05	1.70	1.14E−05	2.33
1.0	1.0	9.38E−02		2.43E−03		5.35E−04		2.39E−03	
	0.5	3.73E−02	1.33	1.34E−03	0.86	5.40E−04	−0.01	1.45E−03	0.72
	0.2	7.02E−03	1.82	1.99E−04	2.08	1.09E−04	1.75	2.43E−04	1.95
	0.1	2.92E−03	1.27	3.32E−05	2.58	2.25E−05	2.28	4.47E−05	2.44
1.2	1.0	2.65E−01		5.65E−03		1.54E−03		5.81E−03	
	0.5	1.17E−01	1.18	3.79E−03	0.58	1.35E−03	0.19	4.17E−03	0.48
	0.2	2.15E−02	1.85	7.36E−04	1.79	3.02E−04	1.63	8.66E−04	1.72
	0.1	8.38E−03	1.36	1.36E−04	2.44	6.90E−05	2.13	1.74E−04	2.32
1.5	1.0	1.31E+00		1.93E−02		1.03E−03		1.93E−02	
	0.5	6.88E−01	0.93	1.38E−02	0.48	4.26E−03	−2.05	1.48E−02	0.38
	0.2	1.05E−01	2.05	5.31E−03	1.04	1.57E−03	1.09	5.87E−03	1.01
	0.1	1.34E−01	−0.35	1.36E−03	1.97	4.57E−04	1.78	1.61E−03	1.87

rors in Table II. The absolute errors are a factor of 332.0716 smaller in Table III because a different unit for electrostatic potential (e_c/Å) is used. In this analysis, errors are computed for different components, such as the flux, ϕ^0, $\tilde{\phi}$, and ϕ. This analysis is important to our understanding of the advantage and disadvantage of MIBPB-III in more complicated cases later. Note that L_∞ errors, which are measured over the whole domain, are reported for ϕ^0 and $\tilde{\phi}$ because they are well behaved functions. First, for a from a=0.2 Å to a=0.6 Å, where the distance between the charge and the interface is greater than 1.4 Å, it is seen that the flux, ϕ^0, $\tilde{\phi}$, and ϕ, can all approximately achieve second order accuracy. For this distance range, most charges are located away from the molecular surface. Second, for a from a=0.8 Å to a=1.2 Å, second order accuracy cannot be achieved for all these components and ϕ. However, if we examine the errors of ϕ^0, $\tilde{\phi}$, and ϕ, it is seen that the errors at h=1.0 Å are very small and very close to those at h=0.5 Å, which results in the low order convergence from h=1.0 Å to h=0.5 Å. This may be an advantage when we use the scheme on a coarse mesh like h=1.0 Å. Third, when a is as large as 1.5 Å, where the distance between the charge and the interface is about 0.5 Å, the order of convergence of the flux, ϕ^0 and $\tilde{\phi}$, is very low. However, even under this situation, the solution, ϕ, is still of about second order accuracy. In short, although the accuracy of the overall solution is limited by its components, it shows a reasonable trend of second order convergence.

Case 3: A dielectric sphere with two off-centered charges. Our third case is a dielectric sphere with two off-centered charges located on the x axis and the z axis of a distance a from the origin. Table IV gives the errors in the electrostatic potential. The distribution of errors observed in this case is very similar to that in Case 2. This study indicates that the number of charges does not have much influence on the accuracy of these numerical schemes. However, the distribution of the charge(s) does have a major impact on the numerical accuracy. The problem becomes more difficult when a is getting larger. In general, the proposed MIBPB-III significantly outperforms two other methods.

Case 4: A dielectric sphere with six off-centered charges. Finally, we consider a sphere with six off-centered charges which are distributed in two patterns. In the first pattern, denoted as Case 4(a), six charges are located at (0.4,0.0,0.0), (0.0,0.8,0.0), (0.0,0.0,1.2), (0.0,0.0,−0.4), (−0.8,0.0,0.0),

TABLE IV. Numerical errors of surface electrostatic potentials for two off-centered charges in a dielectric sphere (Case 3).

		PBEQ		MIBPB-II		MIBPB-III	
a	h	E_1	Order	E_1	Order	E_1	Order
0.2	1.0	5.91*E*+01		1.82*E*+01		3.72*E*−01	
	0.5	4.14*E*+01	0.51	1.02*E*+01	0.84	5.44*E*−02	2.77
	0.2	1.81*E*+01	0.90	7.80*E*−01	2.81	1.17*E*−02	1.68
	0.1	9.14*E*+00	0.99	1.00*E*−01	2.96	3.26*E*−03	1.84
0.4	1.0	6.67*E*+01		2.07*E*+01		4.32*E*−01	
	0.5	4.94*E*+01	0.43	1.16*E*+01	0.83	6.58*E*−02	2.71
	0.2	2.16*E*+01	0.90	1.01*E*+00	2.66	1.50*E*−02	1.61
	0.1	1.09*E*+01	0.99	1.10*E*−01	3.20	4.39*E*−03	1.77
0.6	1.0	7.17*E*+01		2.05*E*+01		4.22*E*−01	
	0.5	5.73*E*+01	0.32	1.27*E*+01	0.69	6.12*E*−02	2.78
	0.2	2.60*E*+01	0.86	1.38*E*+00	2.42	1.65*E*−02	1.43
	0.1	1.33*E*+01	0.97	1.80*E*−01	2.94	5.17*E*−03	1.67
0.8	1.0	7.24*E*+01		1.61*E*+01		4.88*E*−01	
	0.5	6.41*E*+01	0.18	1.04*E*+01	0.63	1.69*E*−01	1.53
	0.2	3.16*E*+01	0.77	2.12*E*+00	1.74	2.13*E*−02	2.26
	0.1	1.69*E*+01	0.90	3.00*E*−01	2.82	5.36*E*−03	1.99
1.0	1.0	6.74*E*+01		5.01*E*+01		1.59*E*+00	
	0.5	6.79*E*+01	−0.01	1.91*E*+01	1.39	5.96*E*−01	1.41
	0.2	3.92*E*+01	0.60	3.58*E*+00	1.83	8.57*E*−02	2.12
	0.1	2.23*E*+01	0.81	5.80*E*−01	2.63	1.46*E*−02	2.55
1.2	1.0	5.81*E*+01		2.04*E*+01		3.86*E*+00	
	0.5	6.70*E*+01	−0.21	3.17*E*+01	−0.63	1.61*E*+00	1.26
	0.2	4.87*E*+01	0.35	5.80*E*+00	1.85	3.00*E*−01	1.83
	0.1	3.10*E*+01	0.65	1.34*E*+00	2.11	5.94*E*−02	2.34
1.5	1.0	3.72*E*+01		7.52*E*+00		1.28*E*+01	
	0.5	5.00*E*+01	−0.43	4.28*E*+01	−2.51	5.40*E*+00	1.25
	0.2	5.95*E*+01	−0.19	2.48*E*+01	0.60	1.98*E*+00	1.09
	0.1	5.15*E*+01	0.21	6.37*E*+00	1.96	5.39*E*−01	1.88

and $(0.0,-1.2,0.0)$. While in the second pattern, denoted as Case 4(b), six charges are located at (0.2,0.2,0.2), (0.5,0.5,0.5), (0.8,0.8,0.8), $(0.2,0.2,-0.2)$, $(0.5,-0.5,0.5)$, and $(0.8,-0.8,-0.8)$. Some of the charges in Case 4(b) are closer to the interface. Table V gives the error analysis for these two charge distributions. The proposed MIBPB-III achieves second order accuracy. It is necessary to point out that at a coarse mesh size $h=1.0$ Å, when the charges are very close to the interface, i.e., within 1.0 Å, the computed potential will not be as accurate as the case where charges are located far from the interface. However, because the van der Waals radii of the partially charged atoms are normally larger than 1.2 Å, such small distance between the charge and the surface will not occur in practical bimolecular simulations.

TABLE V. Numerical errors of surface electrostatic potentials for six charges in a dielectric sphere (Case 4).

Example	h	E_1	Order	ΔG
(a)	1.0	3.25		−3011.12
	0.5	1.72	0.918	−2995.41
	0.2	3.09*E*−1	1.87	−2990.20
	0.1	5.91*E*−2	2.39	−2989.52
(b)	1.0	20.2		−3177.64
	0.5	4.68	2.11	−3159.33
	0.2	5.84*E*−1	2.27	−3122.70
	0.1	9.46*E*−2	2.63	−3123.94

Table V also provides the electrostatic free energies of solvation. Case 4(a) shows a better convergence in the solvation energies as the mesh is refined because its charges are located away from the interface. In contrast, for Case 4(b), larger variations in the solvation energies are observed. However, these variations are smaller than 2% and are acceptable for most simulations. In the following subsection, we test the performance of the proposed method for the combination of geometric and charge singularities.

2. An analytically solvable model

Kirkwood's solution in a dielectric sphere serves as a benchmark test for PB solvers. However, it is also important to validate our method for its performance on complex dielectric interfaces, particularly interfaces with geometric singularities. For this purpose, we consider molecular surfaces,

which are often nonsmooth. One of the difficulties in testing numerical methods is the lack of exact solution for the PBE with complex geometry. Here, we construct an analytical solution for systems with arbitrary singular charges and arbitrary dielectric interfaces. Consider the Poisson equation

$$\nabla \cdot \epsilon(\mathbf{r}) \nabla \phi(\mathbf{r}) + k(\mathbf{r}) = 4\pi \sum_{i=1}^{N_m} q_i \delta(\mathbf{r} - \mathbf{r}_i) \tag{36}$$

with discontinuous $\epsilon(\mathbf{r})$ across the interface Γ, which divides the whole domain into two separated parts, the molecular subdomain Ω^- and the solvent subdomain Ω^+. We set $\epsilon^-=1$ and $\epsilon^+=80$. In the singular source term, q_i is the ith partial charge located at $\mathbf{r}_i$. Here, $k=3\cos x \cos y \cos z$. The problem in Eq. (36) is analytically solvable and its exact solution is

$$\phi^-(x,y,z) = \sum_{i=1}^{N_m} \frac{q_i}{\epsilon_0 \sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2}} + \cos x \cos y \cos z, \tag{37}$$

$$\phi^+(x,y,z) = 0. \tag{38}$$

This is a valuable analytical test case and can be applied to PB solvers to check their accuracy and convergence order. Obviously, it can be modified in many different ways. In particular, it can be trivially modified for the linearized PBE, namely, adding a term, $\kappa^2\phi$, to both sides of Eq. (36) without changing the solution.

In Green's function formulation, the solution can be decomposed into three components, i.e., $\phi=\phi^*+\phi^0+\tilde{\phi}$. Here, we can identify the solution to the Poisson equation with the charge forcing term

$$\phi^* = \sum_{i=1}^{N_m} \frac{q_i}{\epsilon_0 \sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2}}.$$

As such, we have $\phi^0+\tilde{\phi}=\cos x \cos y \cos z$ in Ω^-. Here, ϕ^0 is the solution of the boundary value problem, Eq. (17), and $\tilde{\phi}$ is the solution of Eq. (20). However, due to the additional enforcing term of Eq. (37), the jump conditions of the PBE are not given by Eq. (21) but are the following:

$$[\tilde{\phi}] = \phi^+ - \phi^- = -\cos x \cos y \cos z - \phi^*, \tag{39}$$

$$[\epsilon\tilde{\phi}_\xi] = \epsilon^+ \nabla \phi^+ \cdot \mathbf{n} + \epsilon^- \nabla (-\phi^- + \phi^* + \phi^0) \cdot \mathbf{n}, \tag{40}$$

$$[\epsilon\tilde{\phi}_\eta] = \epsilon^+ \nabla \phi^+ \cdot \mathbf{m} + \epsilon^- \nabla (-\phi^- + \phi^* + \phi^0) \cdot \mathbf{m}, \tag{41}$$

$$[\epsilon\tilde{\phi}_\tau] = \epsilon^+ \nabla \phi^+ \cdot \mathbf{l} + \epsilon^- \nabla (-\phi^- + \phi^* + \phi^0) \cdot \mathbf{l}, \tag{42}$$

where $\mathbf{n}=(p_{11},p_{12},p_{13})^T$, $\mathbf{m}=(p_{21},p_{22},p_{23})^T$, and $\mathbf{l}=(p_{31},p_{32},p_{33})^T$ are the rows from the interface transform matrix $\mathbf{p}$ given by Eq. (26). This benchmark test can be used to validate detailed components of the numerical solution of the proposed MIBPB-III, including its treatment of singular charges and its treatment of interface singularities. In the following two subsections, we apply this test to our methods with different interfaces, including the molecular surfaces of polyatomic systems and the molecular surfaces of proteins.

3. Molecular surfaces of polyatomic systems

We first consider systems with 1, 2, and 18 atoms. For the monoatomic system, a unit charge is located at the atomic center. The coordinates for the diatomic system are $(-2,0,0)$ and $(2,0,0)$, with a unit charge at the center of each atom. Atomic radii are set to 2.0 for these two systems. The molecular surface of the diatomic system is constructed analytically. The atomic coordinates (x,y,z) and van der Waals radius (r) for the 18-atom system are given as $(x,y,z,r)=$(−2.0270 0.9540 −0.6510 1.7), (−1.6690 0.2340 0.6650 1.7), (−0.4530 −0.6870 0.4410 1.7), (0.7510 0.1480 −0.0400 1.7), (0.3930 0.8680 −1.3560 1.7), (−0.8230 1.7880 −1.1320 1.7), (−2.2840 0.2080 −1.4180 1.2), (−2.8880 1.6170 −0.4830 1.2), (−2.5270 −0.3680 0.9970 1.2), (−1.4260 0.9800 1.4350 1.2), (−0.1960 −1.1890 1.3850 1.2), (−0.7010 −1.4410 −0.3200 1.2), (1.0070 0.8940 0.7270 1.2), (1.6120 −0.5150 −0.2080 1.2), (0.1490 0.1210 −2.1260 1.2), (1.2510 1.4700 −1.6880 1.2), (−1.0810 2.2910 −2.0760 1.2), and (−0.5750 2.5430 −0.3710 1.2). The partial charges of these atoms are artificially assigned to the atomic centers as $0.2\ (-1)^i$, $i=1,2,\ldots,18$. The molecular surfaces of these systems are used in Eq. (36) to test the proposed method. Note that the molecular surfaces of the diatomic and 18-atom systems are very irregular. Both MIBPB-II and MIBPB-III have built in rigorous interface treatments for these irregular features on the interface.

Table VI shows both the numerical errors and convergence orders of MIBPB-II and MIBPB-III. E_2 cannot be measured for these cases because the solution vanishes at some grid points. Second order accuracy is obtained by both methods. However, MIBPB-III demonstrates a more consistent convergence pattern and shows a better overall accuracy. Because of the large atomic radius of $r=2.0$ in the monoatomic and diatomic systems, we observe little interference effect of singular charges and the interfaces. Therefore, two methods have the same level of accuracy at $h=1.0$. However, the difference in the performance of two methods at $h=1.0$ is very dramatic in the 18-atom system where some van der Waals radii are as small as $h=1.2$. This difference demonstrates the advantage of the proposed MIBPB-III technique.

4. Molecular surfaces of 24 proteins

It is well known that molecular surface definition admits cusps and self-intersecting singularities.[44] Therefore, it is important to verify that the proposed MIBPB-III is of second order accuracy for the molecular surfaces of proteins. To this end, we consider a set of 24 proteins used in our earlier tests.[60,63] The molecular surfaces of these proteins are used in our tests. For a comparison, our earlier MIBPB-II is also employed. MIBPB-II has built in the rigorous treatment of interface singularities, but without the Green's function treatment for singular charges. Charges are distributed to the neighboring grid points in MIBPB-II, as in many other finite difference based methods.

TABLE VI. Errors and convergence of MIBPB-II and MIBPB-III for solving Eq. (36) with molecular surfaces.

		E_1				E_3			
Systems	MIBPB	1.0	0.5	0.25	0.125	1.0	0.5	0.25	0.125
1 atom	II	2.60*E*−02	9.77*E*−03	1.87*E*−03	2.42*E*−04	8.43*E*−03	2.87*E*−03	3.89*E*−04	4.16*E*−05
	Order		1.41	2.39	2.95		1.55	2.88	3.23
	III	1.18*E*−02	2.00*E*−03	3.27*E*−04	5.04*E*−05	4.90*E*−03	7.05*E*−04	1.12*E*−04	1.61*E*−05
	Order		2.56	2.61	2.70		2.80	2.65	2.80
2 atoms	II	6.21*E*−02	1.63*E*−02	2.49*E*−03	2.54*E*−04	1.78*E*−02	3.63*E*−03	3.30*E*−04	4.18*E*−05
	Order		1.93	2.71	3.29		2.29	3.46	2.98
	III	3.18*E*−02	6.90*E*−03	1.14*E*−03	2.70*E*−04	8.13*E*−03	1.22*E*−03	1.62*E*−04	3.10*E*−05
	Order		2.20	2.60	2.08		2.74	2.91	2.39
18 atoms	II	2.28*E*−01	3.65*E*−02	3.44*E*−03	6.01*E*−04	1.95*E*−02	4.05*E*−03	4.64*E*−04	6.59*E*−05
	Order		2.64	3.41	2.52		2.27	3.13	2.82
	III	4.05*E*−02	9.32*E*−03	1.45*E*−03	2.16*E*−04	7.75*E*−03	1.12*E*−03	1.73*E*−04	2.98*E*−05
	Order		2.12	2.68	2.75		2.79	2.69	2.54

Table VII lists the numerical errors and convergence orders of MIBPB-II and MIBPB-III for the molecular surfaces of 24 proteins. The proteins are listed in the ascending sequence of the radius of gyration. Since numerical errors are collected from the solution at irregular points around the interface, their magnitudes indicate the reliability of PB solvers for the prediction of electrostatic surface potentials. The most important feature in the numerical errors is that MIBPB-III is able to produce much smaller errors at the coarse mesh size of $h=1.0$ Å, which is crucial for the proposed method being used in large scale computations. At $h=1.0$ Å, MIBPB-II is not accurate and reliable due to interference of charge singularities and interface jump conditions. The accuracy of MIBPB-III at $h=1.0$ Å is similar to or even better than that of MIBPB-II at $h=0.50$ Å. At fine mesh resolutions, i.e., $h=0.50$ and 0.25 Å, MIBPB-III is about an order of magnitude more accurate than MIBPB-II. Moreover, MIBPB-III has a consistent order of convergence over two mesh refinements. Whereas, MIBPB-II does not have second order convergence at the last mesh refinement. This reduction in convergence order must be attributed to the presence of the charge singularities because the MIBPB-II was confirmed to have the second order convergence for molecular surface singularities of these 24 proteins.[63]

TABLE VII. The E_1 errors and convergences of MIBPB-II and MIBPB-III for solving Eq. (36) with the molecular surfaces of 24 proteins.

	MIBPB-III					MIBPB-II				
Protein	1.0	0.5	Order	0.25	Order	1.0	0.5	Order	0.25	Order
1ajj	6.52*E*−02	1.85*E*−02	1.82	3.43*E*−03	2.43	8.03*E*−01	7.22*E*−02	3.48	3.18*E*−02	1.18
2pde	8.79*E*−02	2.47*E*−02	1.83	6.77*E*−03	1.87	6.83*E*−01	4.50*E*−02	3.92	3.74*E*−02	0.27
1vii	2.07*E*−01	2.36*E*−02	3.13	4.13*E*−03	2.51	3.71*E*−01	6.37*E*−02	2.54	2.52*E*−02	1.34
2erl	7.36*E*−02	2.17*E*−02	1.76	3.98*E*−03	2.45	1.29*E*+00	6.72*E*−02	4.26	2.26*E*−02	1.57
1cbn	7.52*E*−02	1.60*E*−02	2.23	5.64*E*−03	1.50	6.95*E*−01	7.72*E*−02	3.17	1.94*E*−02	1.99
1bor	9.47*E*−02	2.18*E*−02	2.12	3.92*E*−03	2.48	1.68*E*+00	7.23*E*−02	4.54	1.99*E*−02	1.86
1bbl	1.32*E*−01	2.26*E*−02	2.55	3.42*E*−03	2.72	1.07*E*+00	1.26*E*−01	3.09	3.53*E*−02	1.84
1fca	1.20*E*−01	1.70*E*−02	2.82	2.99*E*−03	2.51	1.33*E*+00	7.83*E*−02	4.09	1.73*E*−02	2.18
1uxc	7.58*E*−02	1.87*E*−02	2.02	3.51*E*−03	2.41	7.82*E*−01	9.01*E*−02	3.12	3.64*E*−02	1.31
1sh1	9.13*E*−02	1.88*E*−02	2.28	5.33*E*−03	1.82	1.71*E*+00	7.34*E*−02	4.54	2.70*E*−02	1.44
1mbg	1.35*E*−01	1.69*E*−02	3.00	3.97*E*−03	2.09	1.78*E*+00	7.92*E*−02	4.49	3.57*E*−02	1.15
1ptq	7.91*E*−02	1.95*E*−02	2.02	4.53*E*−03	2.11	4.42*E*+00	7.92*E*−02	5.80	3.86*E*−02	1.04
1vjw	7.23*E*−02	1.47*E*−02	2.30	3.26*E*−03	2.17	1.23*E*+00	5.92*E*−02	4.38	3.02*E*−02	0.97
1fxd	7.72*E*−02	2.07*E*−02	1.90	3.91*E*−03	2.40	2.14*E*+00	5.07*E*−02	5.40	1.63*E*−02	1.64
1r69	7.96*E*−02	1.84*E*−02	2.11	4.45*E*−03	2.05	1.40*E*+00	9.04*E*−02	3.95	3.24*E*−02	1.48
1hpt	8.08*E*−02	1.92*E*−02	2.07	3.80*E*−03	2.34	2.31*E*+00	9.94*E*−02	4.54	2.63*E*−02	1.92
1bpi	7.33*E*−02	2.11*E*−02	1.80	1.90*E*−03	3.47	5.54*E*+00	8.77*E*−02	5.98	3.35*E*−02	1.39
451c	1.09*E*−01	4.31*E*−02	1.34	8.03*E*−03	2.42	2.50*E*+00	9.27*E*−02	4.75	4.25*E*−02	1.13
1a2s	9.21*E*−02	2.94*E*−02	1.65	5.86*E*−03	2.33	7.48*E*−01	7.86*E*−02	3.25	1.57*E*−02	2.32
1frd	7.95*E*−02	2.26*E*−02	1.81	4.96*E*−03	2.19	1.88*E*+00	8.38*E*−02	4.49	2.96*E*−02	1.50
1svr	8.00*E*−02	2.49*E*−02	1.68	5.02*E*−03	2.31	1.34*E*+00	1.01*E*−01	3.73	4.36*E*−02	1.21
1neq	8.54*E*−02	2.89*E*−02	1.56	5.78*E*−03	2.32	1.90*E*+00	8.89*E*−02	4.42	3.83*E*−02	1.21
1a63	1.86*E*−01	2.16*E*−02	3.11	5.70*E*−03	1.92	2.07*E*+00	8.76*E*−02	4.56	2.51*E*−02	1.80
1a7m	1.17*E*−01	2.88*E*−02	2.02	4.57*E*−03	2.66	1.57*E*+00	8.72*E*−02	4.17	2.49*E*−02	1.81

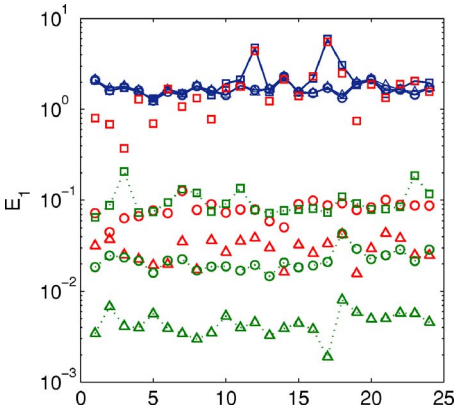


FIG. 5. (Color online) Comparison of absolute maximum errors (E_1) of the surface potentials of 24 proteins, which are ordered with ascending radii of gyration: 1ajj, 2pde, 1vii, 2erl, 1cbn, 1bor, 1bbl, 1fca, 1uxc, 1sh1, 1mbg, 1ptq, 1vjw, 1fxd, 1r69, 1hpt, 1bpi, 451c, 1a2s, 1frd, 1svr, 1neq, 1a63, and 1a7m. Errors of MIBPB-III (green) and PBEQ (blue) at 24 proteins are, respectively, linked with dash lines and solid lines. Errors of MIBPB-II (red) are unlinked. Here, □, ○, and △, are, respectively, for $h=1.0$, 0.5, and 0.25 Å.

Figure 5 depicts the maximum absolute errors (E_1) at three mesh sizes. This figure brings the above comparison in contrast with the errors of PBEQ. Obviously, MIBPB-III errors are a few orders of magnitude smaller than those of PBEQ, which are distributed between 10^0 and 10^1 for all meshes. The PBEQ fails to deliver accurate solutions even if the mesh is further refined because of the discontinuous nature of the solution at molecular surface interfaces.

B. Applications

Having validated the proposed MIBPB-III, we consider its application to the electrostatic free energies of solvation and electrostatic surface potentials.

1. Electrostatic free energies of solvation of proteins

Table VIII lists the details of electrostatic free energies of solvation of 24 proteins. For each method, results are reported at three mesh sizes, $h=1.0$, 0.5, and 0.25 Å. Differences are computed with respect to the results obtained at the finest mesh $h=0.25$ Å, and their magnitudes can be regarded as an indication of convergence. MIBPB-III and MIBPB-II have a similar level of accuracy and convergence at mesh size $h=0.5$ Å. However, MIBPB-II obviously produces larger differences at mesh size $h=1.0$ Å than MIBPB-III. In Table VIII, the CPU time of two methods is compared at $h=0.5$ Å. At a given mesh size, MIBPB-II is faster than MIBPB-III due to the following two reasons. First, in solving the PBE, due to the conversion of the singular charges to the interface jump conditions, the matrix of MIBPB-III has a relatively larger condition number and requires more CPU time. Second, to calculate the electrostatic free energies, MIBPB-II solves an additional Poisson equation by using the FFT scheme, which is very fast. Whereas, MIBPB-III has to solve an additional boundary value problem.

Figure 6 provides a comparison of the MIBPB-III, PBEQ, and APBS. First, there is a good consistency in the solvation energies from three methods, as shown in Fig. 6(a). However, this consistency is illustrated at the scale of

TABLE VIII. Electrostatic free energies of solvation of 24 proteins.

	MIBPB-III					MIBPB-II					CPU	
Protein	0.25	0.5	Diff.	1.0	Diff.	0.25	0.5	Diff.	1.0	Diff.	III	II
1ajj	−1137.2	−1139.9	−2.6	−1133.1	4.1	−1137.2	−1141.2	−4.0	−1160.8	−23.6	68	38
2pde	−820.9	−819.1	1.8	−818.0	2.9	−820.9	−819.4	1.5	−826.7	−5.8	82	46
1vii	−901.2	−900.4	0.8	−896.9	4.3	−901.0	−902.4	−1.4	−910.5	−9.5	67	43
2erl	−948.8	−948.2	0.7	−962.4	−13.6	−948.5	−950.2	−1.6	−992.9	−44.4	62	38
1cbn	−303.8	−303.5	0.3	−309.5	−5.7	−303.7	−305.5	−1.8	−335.5	−31.8	96	53
1bor	−853.7	−858.7	−5.0	−851.7	2.0	−853.8	−860.7	−6.9	−871.8	−18.0	112	61
1bbl	−986.8	−986.5	0.4	−996.7	−9.8	−987.0	−993.5	−6.5	−995.0	−8.0	83	48
1fca	−1200.1	−1199.0	1.1	−1210.4	−10.3	−1200.1	−1201.3	−1.2	−1236.5	−36.4	83	41
1uxc	−1138.7	−1137.2	1.5	−1153.1	−14.5	−1138.9	−1141.4	−2.5	−1164.3	−25.4	93	52
1sh1	−753.3	−752.2	1.0	−742.7	10.6	−753.2	−753.9	−0.7	−777.0	−23.8	82	45
1mbg	−1346.1	−1347.5	−1.4	−1362.6	−16.5	−1346.1	−1350.1	−4.1	−1379.1	−33.0	106	56
1ptq	−873.1	−872.0	1.0	−889	−16.0	−872.7	−871.9	0.8	−889.1	−16.4	96	54
1vjw	−1237.9	−1237.0	0.9	−1247.9	−9.9	−1237.9	−1237.8	0.0	−1277.7	−39.9	109	61
1fxd	−3300.0	−3299.2	0.8	−3316.1	−16.2	−3299.9	−3299.7	0.3	−3347.0	−47.0	103	60
1r69	−1089.5	−1086.2	3.3	−1119.3	−29.8	−1089.2	−1088.2	1.1	−1111.8	−22.6	115	55
1hpt	−814.3	−810.7	3.6	−825.3	−11.1	−813.9	−812.0	1.9	−863.0	−49.1	115	57
1bpi	−1301.9	−1298.1	3.9	−1316.8	−14.9	−1302.0	−1301.9	0.1	−1318.2	−16.2	125	74
451c	−1024.6	−1023.1	1.5	−1023.6	1.0	−1024.6	−1024.9	−0.4	−1051.7	−27.2	156	75
1a2s	−1913.5	−1911.8	1.7	−1923.8	−10.3	−1913.5	−1912.4	1.1	−1963.0	−49.5	189	117
1frd	−2851.9	−2848.5	3.3	−2886.7	−34.8	−2851.1	−2852.6	−1.5	−2932.1	−81.0	210	101
1svr	−1711.2	−1709.0	2.2	−1743.7	−32.5	−1711.2	−1713.5	−2.3	−1799.6	−88.3	228	132
1neq	−1730.1	−1727.7	2.4	−1760.6	−30.5	−1729.9	−1730.9	−1.0	−1780.0	−50.1	227	127
1a63	−2373.5	−2370.6	2.9	−2413.3	−39.8	−2373.4	−2376.0	−2.6	−2458.0	−84.6	454	230
1a7m	−2155.5	−2153.3	2.2	−2173.4	−18.0	−2155.3	−2156.4	−1.1	−2199.4	−44.1	594	316

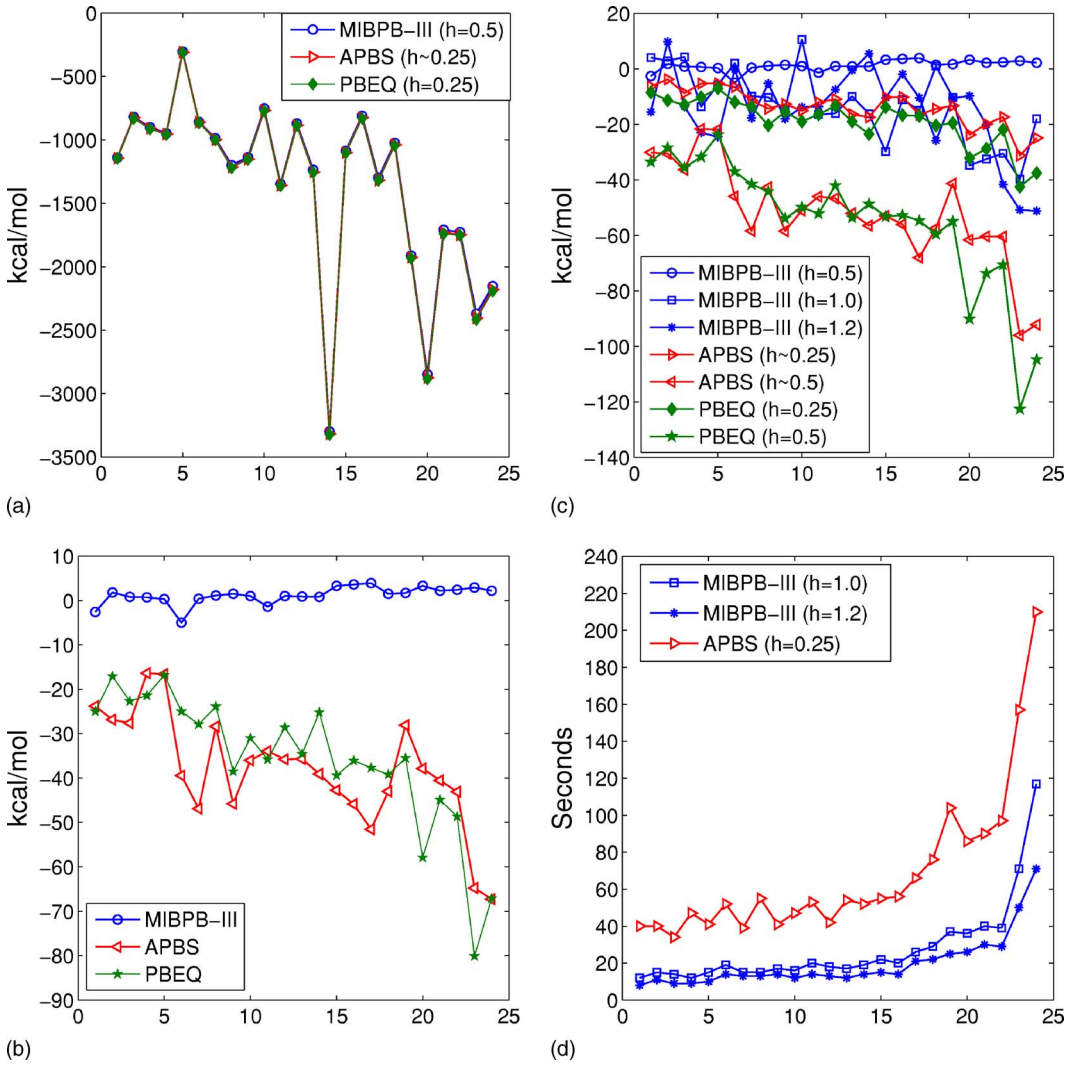


FIG. 6. (Color online) Comparison of electrostatic free energies of solvation of 24 proteins listed in the order of gyration radii. (a) Solvation energies, (b) differences of solvation energies between mesh sizes of $h=0.25$ and $h=0.5$ Å, (c) differences of solvation energies between MIBPB-III at $h=0.25$ Å and the results in the legend, and (d) CPU time.

3500 kcal/mol. In fact, each method has a very different convergence property as shown in Fig. 6(b), where the difference of solvation energies obtained at two different mesh sizes, $h=0.5$ and 0.25, is plotted for each method. MIBPB-III demonstrates an excellent convergence as the mesh is refined. Whereas, results of APBS and PBEQ show large variations over the mesh refinement. As such, it is reasonable to regard the solvation energies of MIBPB-III at mesh size $h=0.25$ Å as converged ones and use them to assess the accuracy of other schemes. In Fig. 6(c), the differences of solvation energies between those obtained by MIBPB-III at mesh size $h=0.25$ Å and others are plotted. Indeed, as the mesh is refined, the solvation energies of APBS and PBEQ converge toward those of MIBPB-III obtained at $h=0.25$ Å. In fact, the magnitudes of variations of MIBPB-III obtained at very coarse mesh sizes of $h=1.0$ and 1.2 Å are in the same range of those of APBS and PBEQ obtained at $h=0.25$ Å. Clearly, APBS and PBEQ at the mesh size of $h=0.5$ Å produce larger differences in solvation energies than does the MIBPB-III at a very coarse mesh sizes of $h=1.0$ and 1.2 Å. This analysis justifies the CPU comparison of MIBPB-III at the mesh size of $h=1.0$ and 1.2 Å with APBS and PBEQ at mesh size of $h=0.25$ for practical electrostatic calculations. As such, we plot the CPU time used by APBS, which is well known for its fast speed, and those used by MIBPB-III in Fig. 6(d). The CPU time is recorded on a Pentium IV personal computer with 2.8 GHz CPU and 2 gbyte random access memory. It is seen that at a similar level of accuracy, MIBPB-III is about three times faster than APBS. However, it is necessary to point out that at the same mesh size, APBS is much faster than MIBPB-III. Table IX provides detailed data for the above comparison and discussion.

2. Electrostatic potentials

We consider the surface electrostatic potential of a heme-binding protein, Fe(II) cytochrome C551 from the organism Pseudomonas aeruginosa (PDB ID: 451c). The heme group is removed and left with a cavity. A common computational domain, $[-17,31]\times[-37,10]\times[-9,36]$, is used for all calculations so that computed electrostatic potentials can be compared on the same set of grids. The first row of Table X lists the maximum and minimum of the electrostatic po-

TABLE IX. Electrostatic free energies of solvation of 24 proteins.

	Solvation energy (kcal/mol)								CPU time (s)		
	MIBPB-III				APBS		PBEQ		MIBPB-III		APBS
h	0.25	0.5	1.0	1.2	0.25	0.5	0.25	0.5	1.0	1.2	0.25
1ajj	−1137.2	−1139.9	−1133.1	−1152.7	−1143.5	−1167.3	−1145.7	−1170.7	12	8	40
2pde	−820.9	−819.1	−818.0	−811.1	−824.6	−851.4	−832.2	−849.3	15	11	40
1vii	−901.2	−900.4	−896.9	−914.8	−909.9	−937.5	−914.1	−936.8	14	9	34
2erl	−948.8	−948.2	−962.4	−971.9	−954.1	−970.5	−959.1	−980.5	12	9	47
1cbn	−303.8	−303.5	−309.5	−328.3	−309.0	−325.6	−310.8	−327.6	15	10	41
1bor	−853.7	−858.7	−851.7	−853.9	−860.2	−899.7	−865.7	−890.7	19	14	52
1bbl	−986.8	−986.5	−996.7	−1004.6	−998.3	−1045.2	−1000.5	−1028.4	15	13	39
1fca	−1200.1	−1199.0	−1210.4	−1205.4	−1214.4	−1242.8	−1220.4	−1244.3	15	13	55
1uxc	−1138.7	−1137.2	−1153.1	−1156.7	−1151.3	−1197.1	−1154.1	−1192.6	17	14	41
1sh1	−753.3	−752.2	−742.7	−767.1	−768.3	−804.3	−772.2	−803.2	16	12	47
1mbg	−1346.1	−1347.5	−1362.6	−1359.7	−1358.2	−1392.1	−1362.5	−1398.3	20	14	53
1ptq	−873.1	−872.0	−889.0	−880.6	−883.9	−919.7	−886.4	−915.0	18	13	42
1vjw	−1237.9	−1237.0	−1247.9	−1238.5	−1254.3	−1290.0	−1256.9	−1291.5	17	12	54
1fxd	−3300.0	−3299.2	−3316.1	−3294.5	−3317.4	−3356.5	−3323.5	−3348.7	19	14	52
1r69	−1089.5	−1086.2	−1119.3	−1100.0	−1099.8	−1142.5	−1103.3	−1142.7	22	15	55
1hpt	−814.3	−810.7	−825.3	−816.2	−824.4	−870.3	−830.9	−867.0	20	14	56
1bpi	−1301.9	−1298.1	−1316.8	−1312.4	−1318.4	−1370.0	−1318.9	−1356.6	26	21	66
451c	−1024.6	−1023.1	−1023.6	−1050.4	−1039.0	−1082.0	−1045.0	−1084.2	29	22	76
1a2s	−1913.5	−1911.8	−1923.8	−1924.1	−1926.8	−1954.9	−1933.0	−1968.5	37	25	104
1frd	−2851.9	−2848.5	−2886.7	−2861.6	−2875.6	−2913.5	−2884.0	−2941.9	36	26	86
1svr	−1711.2	−1709.0	−1743.7	−1731.4	−1731.0	−1771.6	−1740.0	−1784.9	40	30	90
1neq	−1730.1	−1727.7	−1760.6	−1771.7	−1747.5	−1790.5	−1752.0	−1800.7	39	29	97
1a63	−2373.5	−2370.6	−2413.3	−2424.3	−2404.7	−2469.5	−2416.0	−2496.1	71	50	157
1a7m	−2155.5	−2153.3	−2173.4	−2206.7	−2180.4	−2247.6	−2193.0	−2260.1	117	71	210

tential computed by using the MIBPB-III at mesh size $h=0.25$ Å. These values provide an idea about the variability of electrostatic potentials on the mesh. The convergence of MIBPB-III is studied by the differences of the electrostatic potentials computed at mesh sizes $h=0.5$ and 1.0 Å, with respect to that computed at $h=0.25$. There are relatively small deviations in the potentials computed by MIBPB-III. In contrast, the differences of electrostatic potentials of MIBPB-II show a significant deviation. The same feature is also observed for PBEQ. One possible reason for these deviations is that the grids that carry the distributed charges vary as the mesh is refined in these finite difference based methods. The differences of the electrostatic potentials with respect to that obtained by MIBPB-III at $h=0.25$ are listed in the last four rows of Table X. Dramatical variations are observed, indicating the mismatching of the charge distributions in these methods. Note that the charge distribution is exact in MIBPB-III, and the Green's function solution for charges is also exact in MIBPB-III. This finding may have substantial ramification for the reliability of the electrostatic force computed by finite difference based PB solvers. A detailed study of this issue is beyond the scope of this paper.

TABLE X. Electrostatic potential values and differences (kcal/mol/e_c) for cytochrome C551.

Methods	Maximum value	Minimum value
$\phi_{\text{MIBPB-III}(h=0.25)}$	8230.13	−5515.96
$\phi_{\text{MIBPB-III}(h=0.5)}-\phi_{\text{MIBPB-III}(h=0.25)}$	34.54	−38.11
$\phi_{\text{MIBPB-III}(h=1.0)}-\phi_{\text{MIBPB-III}(h=0.25)}$	150.64	−45.68
$\phi_{\text{MIBPB-II}(h=0.5)}-\phi_{\text{MIBPB-II}(h=0.25)}$	980.89	−968.10
$\phi_{\text{PBEQ}(h=0.5)}-\phi_{\text{PBEQ}(h=0.25)}$	984.64	−967.67
$\phi_{\text{MIBPB-II}(h=0.5)}-\phi_{\text{MIBPB-III}(h=0.25)}$	2490.40	−4242.73
$\phi_{\text{MIBPB-II}(h=0.25)}-\phi_{\text{MIBPB-III}(h=0.25)}$	4526.79	−6428.70
$\phi_{\text{PBEQ}(h=0.5)}-\phi_{\text{MIBPB-III}(h=0.25)}$	2491.69	−4242.03
$\phi_{\text{PBEQ}(h=0.25)}-\phi_{\text{MIBPB-III}(h=0.25)}$	4525.75	−6428.72

The electrostatic compatibility of the protein and heme in the cavity region is important to their binding affinity. Figure 7(a) illustrates surface electrostatic mapping of cytochrome C551 computed by MIBPB-III at $h=0.25$ Å. The cavity region is electronically positive except one site, which provides a favorable environment for the heme. The convergence studies of three methods are given in Figs. 7(b)–7(d), obtained by the difference of the potentials between mesh size $h=0.5$ and $h=0.25$ Å for each method. It is seen that MIBPB-III shows an excellent convergence at $h=0.5$ Å, as there is little deviation in two potentials. MIBPB-II exhibits a good convergence. Its deviations are evenly distributed on the interface, indicating that its errors are not created from geometric singularities. However, large deviations at the scale of ±5 kcal/mol/e_c are observed from the potentials computed by PBEQ, which will have an impact in the electrostatic steering effect of the heme binding process.

The differences of surface electrostatic potentials obtained under various conditions and that of MIBPB-III at $h=0.25$ Å are plotted in Fig. 8. It is seen that electrostatic potential obtained by MIBPB-III at $h=1.0$ Å shows consid-

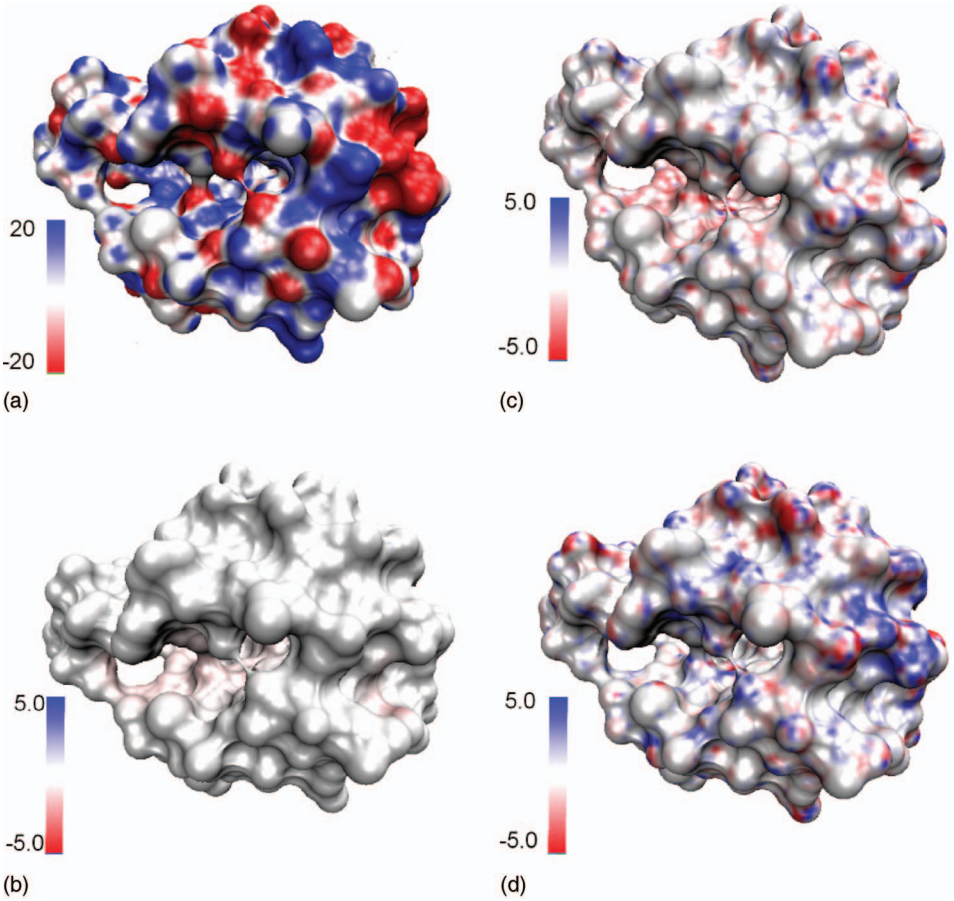


FIG. 7. (Color) Analysis of surface electrostatic potentials of cytochrome C551. (a) $\phi_{\text{MIBPB-III}(h=0.25)}$, and (b) $\phi_{\text{MIBPB-III}(h=0.5)}-\phi_{\text{MIBPB-III}(h=0.25)}$, (c) $\phi_{\text{MIBPB-II}(h=0.5)}-\phi_{\text{MIBPB-II}(h=0.25)}$, and (d) $\phi_{\text{PBEQ}(h=0.5)}-\phi_{\text{PBEQ}(h=0.25)}$.

erable deviations inside the cavity. The deviations obtained from MIBPB-II at $h=0.5$ Å are relatively small and evenly distributed. While the deviations obtained from PBEQ are quite large, particularly for the coarse mesh $h=0.5$ Å.

IV. CONCLUSION

This paper presents a novel Poisson-Boltzmann (PB) solver based on a Green's function formulation and the matched interface and boundary (MIB) method. Our earlier MIB based PB solver, MIBPB-II, has built in an advanced mathematical treatment of dielectric interfaces so that the subgrid information of the interface inside a mesh is accounted and flux continuity conditions at the solvent-molecule interface are rigorously enforced. Moreover, the MIBPB-II maintains its second order accuracy for geometric singularities, including cusps and self-intersecting surfaces in the molecular surfaces,[41] without resorting to surface smoothing procedures. Consequently, the MIBPB-II is more accurate at a coarse mesh than traditional PB solvers at a fine mesh. Nevertheless, MIBPB-II suffers from an accuracy reduction if the mesh size is about half of the van der Waals radius because of the interference of grid points that carry the interface treatment and the grid points that carry the singular charges. The present work effectively overcomes this difficulty by developing a Green's function formalism for charge singularities. The essence is to split the solution into a linear combination of three components in which the singular charge source term is analytically resolved and leads to flux jump conditions at the interface for the PB equation without the singular charge term. The MIB method is then employed to solve the resulting PB equation and treat flux jump conditions from both the original interface and the singular charge contributions on an equal footing. This new PB solver, denoted as MIBPB-III, is extensively validated by benchmark tests, including Kirkwood's dielectric sphere,[67] molecular surfaces of 1, 2, and 18 atoms, and molecular surfaces of 24 proteins. Although molecular surfaces possess geometric singularities, the designed second order accuracy, i.e., numerical error reduces by a factor of 4 when the mesh is halved, is confirmed in the aforementioned tests. It is found that the MIBPB-III provides some of the most accurate solutions to the PBE, to our knowledge. At a mesh as coarse as 1.2 Å, MIBPB-III is able to deliver a similar level of accuracy as other established PB solvers, including PBEQ and APBS, at a fine mesh of 0.25 Å. As such, at the same accuracy, the MIBPB-III is about three times faster than the state of the art multigrid PB solver, the APBS. However, at the same mesh, the MIBPB-III normally requires more CPU time than PBEQ, APBS, and MIBPB-II. Since the Green's function solution to the singular charges is exact, the present treatment provides correct peak values for the electrostatic potential, which may be an advantage in the prediction of electrostatic forces for molecular dynamics. This aspect is under our investigation.

ACKNOWLEDGMENTS

This work was supported in part by NSF Grant Nos. DMS-0616704 and IIS-0430987. The authors thank Yongcheng Zhou for useful discussions.

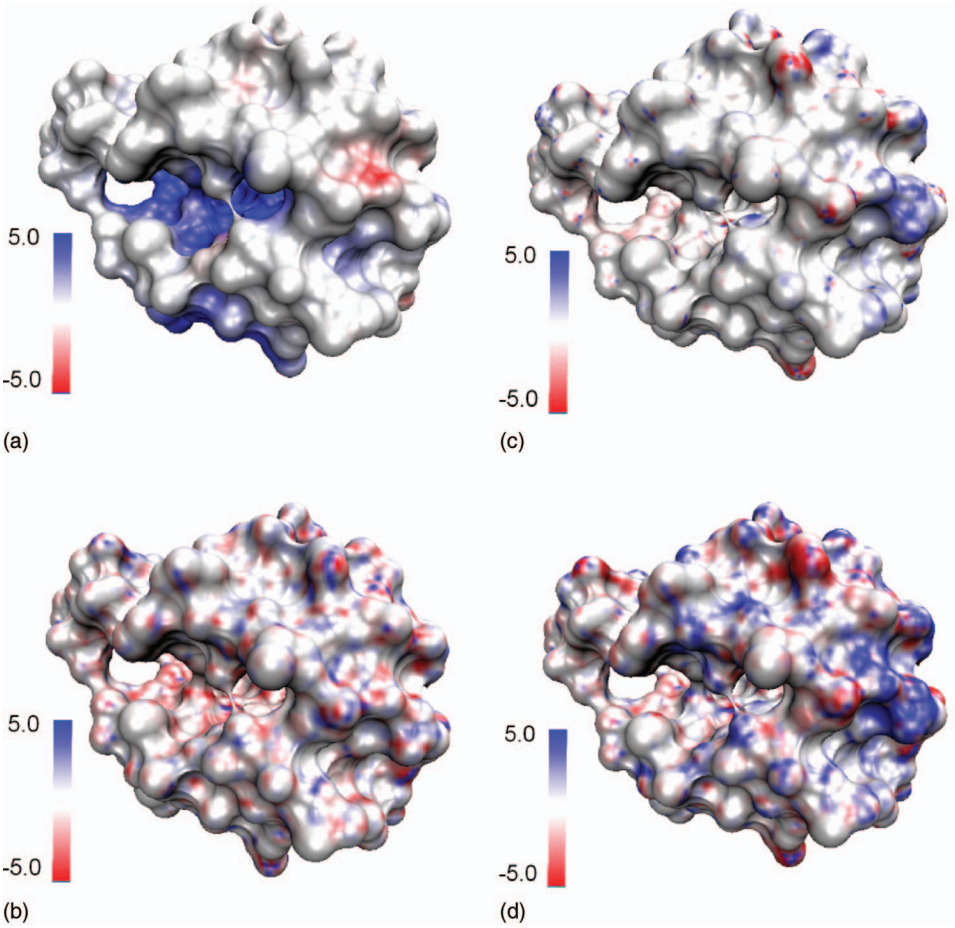


FIG. 8. (Color) Differences of surface electrostatic potentials of cytochrome C551. (a) $\phi_{\text{MIBPB-III}(h=1.0)} - \phi_{\text{MIBPB-III}(h=0.25)}$, and (b) $\phi_{\text{MIBPB-II}(h=0.5)} - \phi_{\text{MIBPB-III}(h=0.25)}$, (c) $\phi_{\text{PBEQ}(h=0.25)} - \phi_{\text{MIBPB-III}(h=0.25)}$, and (d) $\phi_{\text{PBEQ}(h=0.5)} - \phi_{\text{MIBPB-III}(h=0.25)}$.

APPENDIX: KIRKWOOD'S SOLUTION

Kirkwood's dielectric sphere provides an excellent benchmark for testing Poisson-Boltzmann (PB) solvers in terms of accuracy, speed of convergence, and efficiency. In his paper,[67] Kirkwood provided very detailed physical analysis and solutions to the PB equation of a spherical molecule in solvent. Here is a brief summary of Kirkwood's results used in the present work.

The model is a spherical molecule with dielectric boundary (radius a) and ionic boundary (radius b) in solvent. The dielectric constant is ϵ^- inside dielectric boundary and ϵ^+ in the exclusion layer and the solvent domain. The Laplace equation in sphere coordinates is given by

$$\frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2\frac{\partial \phi}{\partial r}\right) + \frac{1}{r^2 \sin\theta}\frac{\partial}{\partial \theta}\left(\sin\theta\frac{\partial \phi}{\partial \theta}\right) + \frac{1}{r^2 \sin^2\theta}\frac{\partial^2 \phi}{\partial \varphi^2} = 0, \tag{A1}$$

where r is the radial coordinate, $\theta \in [0, \pi]$ is the polar angle from the z axis, and $\varphi \in [0, 2\phi)$ is the azimuthal angle in the x-y plane with respect to the x axis.

The solution of Eq. (A1) has the following expansions:

$$\phi = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}(K_{mn}r^{-n-1} + J_{mn}r^n)P_n^m(\cos\theta)e^{im\varphi}, \tag{A2}$$

here P_n^m are the associated Legendre function of the first kind, and K_{mn} and J_{mn} are constants to be determined by boundary or interface conditions.

- Inside the spherical molecule, where singular charges exist, we have

$$-\nabla^2\phi^- = 4\pi\sum_{i=1}^{N_m} q_i\delta(\mathbf{r} - \mathbf{r}_i). \tag{A3}$$

The potential ϕ^-, as the solution to Eq. (A3), can be written as

$$\phi^- = \phi^* + \hat{\phi}, \tag{A4}$$

where ϕ^* is the Green's function

$$\phi^* = \sum_{i=1}^{N_m}\frac{q_i}{\epsilon^-|\mathbf{r} - \mathbf{r}_i|}, \tag{A5}$$

where N_m is the number of charges in the spherical molecule, and q_i and $\mathbf{r}_i$ are the charge and position of the ith atom, respectively. The Green's function can be further expanded as

$$\phi^* = \sum_{n=0}^{\infty}\sum_{m=-n}^{n}\frac{E_{mn}}{\epsilon^- r^{n+1}}P_n^m(\cos\theta)e^{im\varphi}, \tag{A6}$$

where

$$E_{mn} = \frac{(n-|m|)!}{(n+|m|)!}\sum_{i=1}^{N_m} q_i r_i^n P_n^m(\cos\theta_i)e^{im\varphi_i}, \tag{A7}$$

with $(r_i, \theta_i, \varphi_i)$ being the polar coordinates of the position of the ith charge. However, in practical calcula-

tions, this expansion converges slowly for small r. Therefore, it is more convenient to directly evaluate Eq. (A5).

The nonsingular term, $\hat{\phi}$, is given by

$$\hat{\phi} = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} (B_{mn} r^n) P_n^m(\cos\theta) e^{im\varphi}. \tag{A8}$$

- In the spherical shell, i.e., the ion-exclusion layer bounded by the spherical surfaces of radii a and b, the potential ϕ^e governed by the equation $\nabla^2 \phi^e = 0$ is given by

$$\phi^e = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} (C_{mn} r^{-n-1} + G_{mn} r^n) P_n^m(\cos\theta) e^{im\varphi}. \tag{A9}$$

However, this part was not used in our model.

- Outside of the sphere of radius b, the potential ϕ^+ is governed by

$$\nabla^2 \phi^+ - \kappa^2 \phi^+ = 0, \tag{A10}$$

where κ is the ionic strength defined previously, and ϕ^+ is given as

$$\phi^+ = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} (A_{mn} r^{-n-1}) e^{-\kappa r} K_n(\kappa r) P_n^m(\cos\theta) e^{im\varphi}, \tag{A11}$$

where

$$K_n(x) = \sum_{s=0}^{n} \frac{2^s n!(2n-s)!}{s!(2n)!(n-s)!} x^s. \tag{A12}$$

- All the coefficients A_{mn}, C_{mn}, G_{mn}, and B_{nm} are to be determined by boundary conditions and interface condition $\phi^- = \phi^e$, $\epsilon^-(\partial\phi^-/\partial r) = \epsilon^+(\partial\phi^e/\partial r)$, $\phi^e = \phi^+$, $\partial\phi^e/\partial r = \partial\phi^+/\partial r$, and $\phi^+(\infty) = 0$.

In fact, we used a two-domain model in this work—the ion-exclusion layer was not considered. Assuming weak ionic strength, the potential inside the molecule, ϕ^-, with E_{mn} defined as in Eq. (A7), is given by

$$\phi^- = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} \left(\frac{E_{mn}}{\epsilon^- r^{n+1}} + B_{mn} r^n \right) P_n^m(\cos\theta) e^{im\varphi}, \tag{A13}$$

and the potential outside the molecule, ϕ^+, is given by

$$\phi^+ = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} (C_{mn} r^{-n-1} + G_{mn} r^n) P_n^m(\cos\theta) e^{im\varphi}. \tag{A14}$$

All coefficients B_{mn}, C_{mn}, and G_{mn} can be obtained via boundary and interface conditions $\phi^- = \phi^+$, $\epsilon^-(\partial\phi^-/\partial r) = \epsilon^+(\partial\phi^+/\partial r)$, and $\phi^+(\infty) = 0$.

Kirkwood's solution provides an effective benchmark for testing our scheme. As mentioned before, the accuracy of the entire scheme for ϕ depends on the accuracy of ϕ^0 solved from the boundary value problem, of the flux computed by differentiating ϕ^0, and of $\tilde{\phi}$ solved from the interface problem. In Kirkwood's solution, ϕ^* in Eq. (A6) corresponds to ϕ^* in Eq. (16). While $\hat{\phi}$ in Eq. (A8) corresponds to the sum of ϕ^0 and $\tilde{\phi}$ in Eq. (14). Finally, ϕ^+ in Eq. (A14) corresponds to the $\tilde{\phi}$ in Eq. (20) for the potential outside of the molecule. Meanwhile, since ϕ^0 satisfies Eq. (17), it will have the following analytical expansion with spherical harmonics:

$$\phi^0 = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} (D_{mn} r^n) P_n^m(\cos\theta) e^{im\varphi}, \tag{A15}$$

where $D_{nm} = E_{nm}/\epsilon^- b^{2n+1}$. The existence of the analytic solution of ϕ^*, ϕ^0, and $\tilde{\phi}$ makes it possible for us to test the accuracy of our scheme step by step in a sphere with multiple charges.

[1] P. Koehl, Curr. Opin. Struct. Biol. **16**, 142 (2006).
[2] C. L. Vizcarra and S. L. Mayo, Curr. Opin. Struct. Biol. **9**, 622 (2005).
[3] F. Fogolari, A. Brigo, and H. Molinari, J. Mol. Recognit. **15**, 377 (2002).
[4] N. A. Baker, Curr. Opin. Struct. Biol. **15**, 137 (2005).
[5] M. Feig and C. L. Brooks III, Curr. Opin. Struct. Biol. **14**, 217 (2004).
[6] C. M. Cortis and R. A. Friesner, J. Comput. Chem. **18**, 1591 (1997).
[7] K. E. Forsten, R. E. Kozack, D. A. Lauffenburger, and S. Subramaniam, J. Phys. Chem. **98**, 5580 (1994).
[8] B. Honig and A. Nicholls, Science **268**, 1144 (1995).
[9] B. Roux and T. Simonson, Biophys. Chem. **78**, 1 (1999).
[10] K. A. Sharp and B. Honig, Annu. Rev. Biophys. Biophys. Chem. **19**, 301 (1990).
[11] Y. N. Vorobjev and H. A. Scheraga, J. Comput. Chem. **18**, 569 (1997).
[12] R. J. Zauhar and R. S. Morgan, J. Mol. Biol. **186**, 815 (1985).
[13] J. Warwicker and H. C. Watson, J. Mol. Biol. **154**, 671 (1982).
[14] F. Fogolari, A. Brigo, and H. Molinari, Biophys. J. **85**, 159 (2003).
[15] P. A. Kollman, I. Massova, and C. Reyes *et al.* Acc. Chem. Res. **33**, 889 (2000).
[16] J. M. J. Swanson, R. H. Henchman, and J. A. McCammon, Biophys. J. **86**, 67 (2004).
[17] R. E. Georgescu, E. G. Alexov, and M. R. Gunner, Biophys. J. **83**, 1731 (2002).
[18] J. E. Nielsen and J. A. McCammon, Protein Sci. **12**, 313 (2003).
[19] J. Warwicker, Protein Sci. **13**, 2793 (2004).
[20] Q. Lu and R. Luo, J. Chem. Phys. **119**, 11035 (2003).
[21] N. V. Prabhu, P. Zhu, and K. A. Sharp, J. Comput. Chem. **25**, 2049 (2004).
[22] B. Z. Lu, W. Z. Chen, C. X. Wang, and X. J. Xu, Proteins **48**, 497 (2002).
[23] R. Luo, L. David, and M. K. Gilson, J. Comput. Chem. **23**, 1244 (2002).
[24] W. Im, D. Beglov, and B. Roux, Comput. Phys. Commun. **111**, 59 (1998).
[25] N. A. Baker, D. Sept, S. Joseph, M. J. Holst, and J. A. McCammon, Proc. Natl. Acad. Sci. U.S.A. **98**, 10037 (2001).
[26] N. A. Baker, D. Sept, M. J. Holst, and J. A. McCammon, IBM J. Res. Dev. **45**, 427 (2001).
[27] M. Holst and F. Saied, J. Comput. Chem. **14**, 105 (1993).
[28] A. Bordner and G. Huber, J. Comput. Chem. **24**, 353 (2003).
[29] A. Boschitsch, M. Fenley, and H.-X. Zhou, J. Phys. Chem. B **106**, 2741 (2002).
[30] A. Boschitsch and M. Fenley, J. Comput. Chem. **25**, 935 (2004).
[31] A. Juffer, E. Botta, B. van Keulen, A. van der Ploeg, and H. Berendsen, J. Comput. Phys. **97**, 144 (1991).
[32] J. Liang and S. Subranmaniam, Biophys. J. **73**, 1830 (1997).
[33] B. Z. Lu, X. L. Cheng, J. F. Huang, and J. A. McCammon, Proc. Natl. Acad. Sci. U.S.A. **103**, 19314 (2006).
[34] I. Klapper, R. Hagstrom, R. Fine, K. Sharp, and B. Honig, Proteins **1**, 47 (1986).
[35] W. Rocchia, E. Alexov, and B. Honig, J. Phys. Chem. B **105**, 6507 (2001).
[36] M. E. Davis, J. D. Madura, B. A. Luty, and J. A. McCammon, Comput. Phys. Commun. **62**, 187 (1991).
[37] M. Holst, N. Baker, and F. Wang, J. Comput. Chem. **21**, 1319 (2000).
[38] N. Baker, M. Holst, and F. Wang, J. Comput. Chem. **21**, 1343 (2000).

[39] B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan, and M. Karplus, J. Comput. Chem. **4**, 187 (1983).
[40] M. L. Connolly, J. Appl. Crystallogr. **18**, 499 (1985).
[41] B. Lee and F. M. Richards, J. Mol. Biol. **55**, 379 (1971).
[42] F. Eisenhaber and P. Argos, J. Comput. Chem. **14**, 1272 (1993).
[43] V. Gogonea and E. Osawa, Supramol. Chem. **3**, 303 (1994).
[44] M. F. Sanner, A. J. Olson, and J. C. Spehner, Biopolymers **38**, 305 (1996).
[45] C. S. Peskin, J. Comput. Phys. **25**, 220 (1977).
[46] R. P. Fedkiw, T. Aslam, B. Merriman, and S. Osher, J. Comput. Phys. **152**, 457 (1999).
[47] X. D. Liu, R. P. Fedkiw, and M. Kang, J. Comput. Phys. **160**, 151 (2000).
[48] I. Babuška, Computing **5**, 207 (1970).
[49] J. Bramble and J. King, Adv. Comput. Math. **6**, 109 (1996).
[50] Z. L. Li, T. Lin, and X. H. Wu, Numer. Math. **96**, 61 (2003).
[51] W. K. Liu, Y. Liu, D. Farrell *et al.*, Comput. Methods Appl. Mech. Eng. **195**, 1722 (2006).
[52] W. Cai and S. Z. Deng, J. Comput. Phys. **190**, 159 (2003).
[53] F. Gibou and R. P. Fedkiw, J. Comput. Phys. **202**, 577 (2005).
[54] A. Mayo, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **21**, 285 (1984).
[55] R. J. LeVeque and Z. L. Li, SIAM (Soc. Ind. Appl. Math.) J. Numer. Anal. **31**, 1019 (1994).
[56] Z. H. Qiao, Z. L. Li, and T. Tang, J. Comput. Math. **24**, 252 (2006).
[57] S. Zhao and G. W. Wei, J. Comput. Phys. **200**, 60 (2004).
[58] Y. C. Zhou, S. Zhao, M. Feig, and G. W. Wei, J. Comput. Phys. **213**, 1 (2006).
[59] Y. C. Zhou and G. W. Wei, J. Comput. Phys. **219**, 228 (2006).
[60] Y. C. Zhou, M. Feig, and G. W. Wei, J. Comput. Chem. (to be published).
[61] S. Hou and X.-D. Liu, J. Comput. Phys. **202**, 411 (2005).
[62] S. N. Yu, Y. C. Zhou, and G. W. Wei, J. Comput. Phys. **224**, 729 (2007).
[63] S. N. Yu, W. H. Geng, and G. W. Wei, J. Chem. Phys. **126**, 244108 (2006).
[64] Z. Zhou, P. Payne, M. Vasquez, N. Kuhn, and M. Levitt, J. Comput. Chem. **17**, 1344 (1996).
[65] I.-L. Chern, J.-G. Liu, and W.-C. Wang, Methods Appl. Anal. **10**, 309 (2003).
[66] S. N. Yu and G. W. Wei, J. Comput. Phys. (to be published).
[67] J. G. Kirkwood, J. Chem. Phys. **7**, 351 (1934).
[68] M. Holst, Ph.D. thesis, University of Illinois at Urbana-Champaign, 1993.
[69] M. Feig, A. Onufriev, M. S. Lee, W. Im, D. A. Case, and C. L. Brooks III, J. Comput. Chem. **25**, 265 (2004).
[70] A. D. MacKerell, Jr., D. Bashford, M. Bellott *et al.*, J. Phys. Chem. **102**, 3586 (1998).