Communications in Information and Systems
Volume 0, Number 0, 1–9, 2022

# Comparison of three matched interface and boundary (MIB) schemes for solving the nonlinear Poisson-Boltzmann equation

Yiming Ren, Sylvia Amihere, Weihua Geng , and Shan Zhao[*],

This study addresses the challenge posed by the nonlinearity of the Poisson-Boltzmann (PB) equation through the lens of the Matched Interface and Boundary (MIB) method in the two-component regularization setting. The regularized Matched Interface and Boundary (rMIB) scheme, which utilizes the inexact-Newton's method to solve the nonlinear PB equation with second order accuracy, serves as the benchmark for our comparison. Inspired by the augmented MIB (AMIB) schemes for linear elliptic problems, an augmented formulation has been formed for the regularized PB equation, which leads to a nonlinear algebraic system. Two fast Poisson solvers have been applied to solve the augmented system. One utilizes the fast Fourier transform (FFT) algorithm in a relaxation process, while the other employs the geometric multigrid method couples with an inexact-Newton's scheme, giving rise to the AMIB-FFT and AMIB-Multigrid schemes. Numerical experiments involving a nonlinear Kirkwood sphere and several proteins have been conducted to assess the accuracy and efficiency of three MIB schemes for solving the nonlinear PB equation.

## 1. Introduction

The Poisson-Boltzmann equation (PBE) is an elliptic equation with singular sources and discontinuous coefficients across an interface. This equation is widely used as the governing equation of the electrostatic interactions of solvated biomolecules in a solvent environment with dissolved electrolytes [7, 14, 11]. In practice, the PBE is solved numerically as its analytical solution is only available for simple shapes such as a sphere in a linearized setting. However, there are several numerical difficulties in solving the PBE, such as complex molecular surface, discontinuous coefficients across the surface,

arXiv: 0000.0000

[*] Corresponding author. ORCID: 0000-0002-3023-2107.

nonlinearity, singular sources, unbounded domain, etc. In this study, we focus on fast Poisson solvers in treating the nonlinearity.

The discretized form of the nonlinear PBE can be solved by several numerical methods, such as the nonlinear relaxation methods as implemented in Delphi and PBEQ [19], the nonlinear conjugate gradient method implemented in UHBD [15], the inexact-Newton method implemented in APBS [16], and the nonlinear multigrid method [18]. The relaxation methods, as an extension of classical linear techniques like Gauss-Seidel and successive overrelaxation, were initially applied to solve the nonlinear PBE [17] using the finite difference discretization. However, these methods may encounter convergence difficulties. On the contrary, the inexact-Newton method has been proven to converge reliably. Compared with traditional Newton's method, the inexact-Newton method offers enhanced efficiency [16]. We note that pseudo-time methods have also been developed [9, 1], in which the nonlinear PBE can be converted into a time-dependent form by introducing a pseudo-time derivative, and the solution is retrieved from the steady-state solution. In this process, the nonlinearity can be perfectly handled when the time-dependent PBE is split into linear and nonlinear subsystems with the latter being analytically integrated [9]. In this study, our focus on employing the relaxation scheme and the inexact-Newton method for the treatment of the nonlinearity.

The objective of this study is to conduct a comprehensive comparison of three finite difference methods within the framework of the Matched Interface and Boundary (MIB) method [24]. The MIB PBE solver [4] is the first known numerical algorithm in the literature that can maintain a second order of accuracy in treating geometric singularities of molecular surfaces [22] and charge singularities [8]. This scheme is further improved with an two-component regularization for the treatment of charge singularities [10], namely the rMIB scheme. The rMIB package is chosen as the benchmark solver in the present study. Such a selection is motivated by its efficiency in solving the discretized linear algebraic system $Ax = b$ through the use of the row-indexed sparse storage mode, and its ability to handle the nonlinearity with the inexact-Newton's method.

Recently, several augmented MIB (AMIB) schemes have been developed for solving linear elliptic problems [5, 6, 20]. Based on similar interface and boundary treatments as in the classical MIB schemes [24, 22, 8, 23, 4, 10], auxiliary variables are introduced in the AMIB schemes to form an augmented linear system so that the standard finite difference discretization of the Laplacian could be preserved. Consequently, fast Poisson solvers, such as the fast Fourier transform (FFT) and geometric multigrid, can be applied in the Schur complement solution of the augmented system. The AMIB schemes thus become more efficient than the classical MIB schemes in solving linear elliptic problems [5, 6, 20]. This motivates us to introduce an augmented MIB formulation for the nonlinear PBE with two-component regularization so that fast Poisson solvers could be applied. To deal with the PBE nonlinearity, a relaxation scheme or inexact-Newton scheme will be utilized, serving as an outer iteration. Within each iteration, a linear augmented system will be solved by using the FFT and geometric multigrid method, respectively, in the relaxation and inexact-Newton iterations, giving rising to the proposed AMIB-FFT and AMIB-Multigrid methods. Based on the same MIB interface treatment and two-component regularization, accuracy and efficiency comparison of three MIB schemes, i.e., rMIB, AMIB-FFT, and AMIB-Multigrid, will be conducted.

The rest of this paper is structured as follows. The proposed numerical approaches are discussed in Section 2. A numerical comparison of the electrostatic potential and energy is reported in Section 3. Finally, this paper ends with a conclusion.

## 2. Theory and algorithm

In the Poisson-Boltzmann (PB) model, $\Omega^-$ and $\Omega^+$ represent the molecule and solvent domains, respectively. These two domains are separated by the molecular surface $\Gamma = \Omega^- \cap \Omega^+$. Note that in the physical model, the entire system is considered in an infinite domain $R^3$. However, for grid-based methods, a bounded computational domain, such as a box $\Omega = \Omega^- \cup \Omega^+$, is applied. The molecular subdomains of the PB model are illustrated in Fig.1.

The PB model governs the electrostatic potential $\phi(r)$ on $\Omega$ by Gauss's law as

$$(1) \qquad \begin{cases} -\epsilon^- \Delta\phi(r) = \rho(r), & r \in \Omega^- \\ -\epsilon^+ \Delta\phi(r) + \bar{\kappa}^2 \sinh\phi(r) = 0, & r \in \Omega^+ \\ [\phi]_\Gamma = 0 & r \in \Gamma \\ \left[\epsilon \frac{\partial \phi}{\partial n}\right]_\Gamma = 0 & r \in \Gamma \\ \phi(r) = \phi_b(r), & r \in \partial\Omega, \end{cases}$$

where $r = (x, y, z)^T \in \mathbb{R}^3$. Here, $\epsilon^+$ and $\epsilon^-$ are the dielectric constants in the solvent region and molecule region respectively. We choose $\epsilon^+ = 80$ and $\epsilon^- = 1$. The ionic screening coefficient $\bar{\kappa}^2$ vanishes in $\Omega^-$ and it is a constant

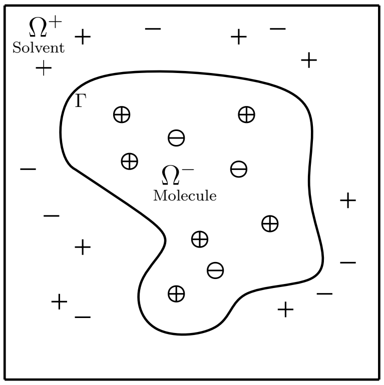

Figure 1: The subdomain setting used in the PB model.

depending on the ionic strength $I$ of the solvent in $\Omega^+$. The singular source term $\rho(r) = 4\pi C \sum_{j=1}^{N_c} q_i \delta(r - r_i)$, where $q_i$ is the partial charge on the $i$th atom center $r_i$ and $C$ is a constant to balance the units [10].

Additionally, $[\phi]_\Gamma = \phi^+ - \phi^-$ and $[\epsilon \phi_n]_\Gamma = \epsilon^+ \frac{\partial \phi^+}{\partial n} - \epsilon^- \frac{\partial \phi^-}{\partial n}$ are the interface conditions across the molecular surface with $n$ as the normal direction. Physically, the potential is defined over the infinite domain with the radiation condition $\lim_{|r| \to \infty} \phi(r) = 0$. However, for grid-based numerical methods, a Dirichlet boundary condition is usually assumed on the boundary of the finite domain $\Omega$. A commonly used Dirichlet condition is given as

$$\phi(r) = \phi_b(r) = C \sum_{i=1}^{N_c} \frac{q_i e^{-|r-r_i|\sqrt{\frac{\bar{\kappa}^2}{\epsilon^+}}}}{\epsilon^+ |r - r_i|}. \tag{2}$$

To handle the inherent charge singularities in the PBE, Geng and Zhao [10] devised a regularization method that involves splitting the PBE solution into two components. Notably, this decomposition occurs exclusively within the molecule region. The potential solution is introduced with the following decomposition:

$$\phi = \phi_{RF} + \mathcal{G}, \text{ in } \Omega^-.$$

Here, $\phi_{RF}$ represents the reaction field potential, and $\mathcal{G}$ is the Green's function defined as:

$$\mathcal{G}(r) = C \sum_{i=1}^{N_c} \frac{q_i}{\epsilon^- |r - r_i|}.$$

With this decomposition in place, the regularized PB equation can be formulated as:

$$\begin{cases} -\epsilon^- \Delta u = 0, & r \in \Omega^- \\ -\epsilon^+ \Delta u + \bar{\kappa}^2 \sinh(u) = 0, & r \in \Omega^+ \\ [u]_\Gamma = \phi(r^+) - \phi_{RF}(r^-) = \mathcal{G}, & r \in \Gamma \\ \left[\epsilon \frac{\partial u}{\partial n}\right]_\Gamma = \epsilon^+ \frac{\partial \phi}{\partial n}(r^+) - \epsilon^- \frac{\partial \phi_{RF}}{\partial n}(r^-) = \epsilon^- \frac{\partial \mathcal{G}}{\partial n}, & r \in \Gamma \\ u = \phi_b, & r \in \partial\Omega, \end{cases} \tag{3}$$

with the regularized potential $u$ defined as

$$u = \begin{cases} \phi_{RF} & \text{in } \Omega^- \\ \phi & \text{in } \Omega^+. \end{cases} \tag{4}$$

Physically, the reaction field potential $\phi_{RF}$ can be interpreted as electrostatic field generated by the charges induced by transferring the environment surrounding the solute from $\epsilon^+$ to $\epsilon^-$. Numerically, one just needs to solve one PB interface problem given in (3). With the solution $u$, the original potential is recovered as $\phi = u$ in $\Omega^+$ and $\phi = u + G$ in $\Omega^-$. Note that the Green's function $G$ and its gradient are analytically known in (3).

In [10], a MIB finite difference algorithm is proposed for solving linear and nonlinear PB equations based on the regularization (3). This rMIB package involves two-layer iterations for solving the nonlinear PB equation. The inexact-Newton iteration is applied as the outer layer, while the Biconjugate Gradient (BCG) method is employed to solve the finite difference linear system in the inner layer. The rMIB is efficient due to several factors. First, the linear systems in the inner iteration are solved inexactly so that the BCG iteration numbers are usually very small, such as one, for most of Newton steps. Second, the outer Newton iteration converges quadratically, so that the total Newton iteration number is also small. Finally, in each BCG step, the matrix vector multiplication can be computed efficiently, because the Laplacian discretization matrix is saved in the row-indexed sparse storage mode [10].

In this investigation, we explored the effectiveness of an augmented MIB (AMIB) approach in handling the nonlinear PB equation through regularization. Two prominent methodologies, the relaxation scheme, and the inexact-Newton method, are considered for tackling the challenges posed by the nonlinearity in the PB equation. Either the relaxation scheme or the inexact-Newton method is integrated as outer iterations within the augmented framework. Thus, a nonlinear augmented system is transformed into a new linear augmented system, which is then solved during the inner iteration. Computational efficiency is improved by employing either the FFT in the inner iteration of the relaxation scheme or the geometric multigrid algorithm in the inner iteration of the inexact-Newton scheme.

### 2.1. An extended domain with an immersed boundary

In finite difference PB solvers, a uniform mesh is usually employed with the spacing $h$ being the same in all Cartesian directions, i.e., $h = \Delta x = \Delta y = \Delta z$, while the number of grid nodes in each direction, $n_x$, $n_y$, and $n_z$, are random integers for real protein systems. In order to apply the FFT or geometric multigrid method, we have to extend the domain $\Omega$ by embedding it in an enlarged cubic domain $D$ with a power-of-2 grid structure for $n_x$, $n_y$, and $n_z$.

We denote the extended domain as $\Omega^e$, and define the interface as $\Gamma_1 = \Omega \cap \Omega^e$, which constitutes the entire domain $D = \Omega \cup \Omega^e$. Solutions within $\Omega^e$ are considered to be zero. Consequently, we can reformulate the modified PBE introduced in (3) during each iteration step as follows:

$$\Delta u - \frac{\bar{\kappa}^2}{\epsilon^+}\sinh(u) = 0, \quad r \in D \tag{5}$$

Here, $\bar{\kappa}^2$ is re-defined as

$$\bar{\kappa}^2 = \begin{cases} 0, & r \in \Omega^- \\ 8.430325455I, & r \in \Omega^+ \\ 0, & r \in \Omega^e \end{cases} \tag{6}$$

The new immersed boundary problem is subject to the original interface conditions on $\Gamma$ and the boundary condition on the immersed interface $\Gamma_1$. On the extended boundary $\partial D$, a trivial zero boundary condition can be assumed.

To address the solution discontinuity near $\Gamma$ and $\Gamma_1$, specific numerical treatments are essential. Within the MIB method, this challenge is mitigated by incorporating crucial irregular points and fictitious values. In essence, while the standard second-order central difference can be applied for approximation at regular points, adjustments are necessary for irregular points where finite difference is not well-defined due to the discontinuous solution across $\Gamma$ and $\Gamma_1$.

To achieve a second-order central difference approximation at all irregular points near $\Gamma$, two layers of fictitious points are required along each Cartesian grid line, surrounding $\Gamma$. The determination of these fictitious values adheres rigorously to jump conditions [10]. Similarly, only one layer of fictitious points outside $\Gamma_1$ is needed along each Cartesian grid line, and these fictitious values are determined based on boundary conditions using the MIB boundary closure approach introduced in [23].

### 2.2. Second order corrected central differences

In the AMIB methods [5, 6, 20], to address those irregular points, we correct the standard second-order central difference at various interface points, which are the intersection points between Cartesian grid lines and $\Gamma$ or $\Gamma_1$. Specifically, we assume that the potential $u$ is a piecewise smooth function in $C^4[x_i - h, \alpha) \cup C^4(\alpha, x_{i+1} + h]$, with $x_i \leq \alpha \leq x_{i+1}$, and the derivatives extend continuously up to $\alpha$. Note that the $C^4$ regularity is usually required in the central difference approximation of second order derivatives for guaranteeing a second order local truncation error. With a discontinuity, the central difference needs to be corrected across the interface. Employing second-order corrected central differences, the Laplacian approximation in $x$ direction can be expressed as:

(7)
$$u_{xx}(x_i) = \frac{u(x_{i-1}) - 2u(x_i) + u(x_{i+1})}{h^2} - \frac{1}{h^2} \sum_{m=0}^{k} \frac{(h^+)^m}{m!}[u^{(m)}] + O(h^{k-1}),$$

(8)
$$u_{xx}(x_{i+1}) = \frac{u(x_i) - 2u(x_{i+1}) + u(x_{i+2})}{h^2} + \frac{1}{h^2} \sum_{m=0}^{k} \frac{(h^-)^m}{m!}[u^{(m)}] + O(h^{k-1}).$$

where $h^- = x_i - \alpha$, $h^+ = x_{i+1} - \alpha$, and $[u^{(m)}]$ are the Cartesian derivative jumps defined at $\alpha$, measuring the difference in the $m$th derivative of $u$ between the right and left sides. By taking $k = 2$, the Laplacian approximations (7) and (8) have a local truncation error of $O(h)$, which can guarantee a global second order convergence. Expanding on this corrected difference in the $x$-direction, we can generalize the concept to three dimensions (3D) by combining corrected differences from different directions to approximate the Laplacian operator dimension by dimension. Throughout all corrected finite differences, the standard central difference is preserved, and the correction terms involve jump quantities at interface points. At regular points away from interfaces, the correction terms vanish, simplifying the corrected finite difference to the standard central difference. Consequently, the corrected finite difference approach is applicable to all grid nodes. Moreover, interested readers are directed to Ref.[12] for the corrected central differences in the corner situations.

### 2.3. Construction of derivative jumps

We next consider how to numerically approximate Cartesian derivative jumps at $\alpha$, i.e., $[u^{(m)}]_\alpha (m = 0, 1, 2)$ with the aid of MIB fictitious values.

Instead of a numerical approximation, the zeroth derivative jump is explicitly obtained by the analytical jump condition. In order to reconstruct other jump values for $m$ up to 2 in the corrected difference, two Lagrange polynomials of degree two can be built to give limiting derivatives, resulting in the approximated jumps up to second order derivative. We can combine two real function values together with one fictitious value on the other side of the interface to obtain the Lagrange polynomial for one-sided subdomain.

By taking derivatives on these one-sided polynomials, the jump values at an intersection point $\alpha$ with regard to the $x$ direction will take a general form in 3D as below

$$\begin{aligned}[\frac{\partial^m u}{\partial x^m}] \approx\ & (w^+_{m,1} f_{i,j,k} + w^+_{m,2} u_{i+1,j,k} + w^+_{m,3} u_{i+2,j,k}) \\ & - (w^-_{m,1} u_{i-1,j,k} + w^-_{m,2} u_{i,j,k} + w^-_{m,3} f_{i+1,j,k})\end{aligned} \tag{9}$$

where $m = 1, 2$. Fictitious values $f_{i,j,k}$ and $f_{i+1,j,k}$ are generated by either the MIB scheme [10] or the MIB boundary closure approach [6], and $w^-_{m,n}$ and $w^+_{m,n}$ denote the weights on each function value after taking a certain derivative of the Lagrange polynomial at $\alpha$. The subscripts signify the order of the function value from the left. Furthermore, the detail of the reconstruction of Cartesian derivative jumps in the corner situations is referred to [20].

### 2.4. Augmented system

Introducing derivative jumps as auxiliary variables, the MIB fictitious value representation is integrated into Eq. (9), yielding a generic linear equation:

$$\sum_{(x_I,y_J,z_K)\in\mathbb{S}i,j,k} C_{I,J,K}u_{I,J,K} + [\frac{\partial^m u}{\partial x^m}] = C_0. \tag{10}$$

Here, $C_{I,J,K}$ are weights corresponding to the function value $u_{I,J,K}$ approximating the jump quantity $[\frac{\partial^m u}{\partial x^m}]$, and $C_0$ is the known jump data. Similar formulas to Eq. (10) are derived for all interface points in $x$-, $y$-, and $z$-directions.

Furthermore, let us denote the 1D column vector $Q$ of dimension $3N_2 \times 1$ as a vector of introduced auxiliary variables $[\frac{\partial^m u}{\partial x^m}]$ at a total of $N_2$ intersection points between grid lines and interfaces. Unknown function values at $N_1$ interior grids within the domain $D$ are organized in a 1D column vector $U$ of dimension $N_1 \times 1$. Generalizing Eq.(10) from one interface point to all interface points, the matrix form of Eq.(10) is then expressed as:

$$CU + IQ = \Phi, \tag{11}$$

where $C$ is a sparse matrix of dimension $3N_2 \times N_1$, $I$ is the identity matrix of dimension $3N_2 \times 3N_2$, and $\Phi$ is a column vector of dimension $3N_2 \times 1$ composed of known interface quantities.

Now, denoting $u_{i,j,k}$ as the discrete solution at $(x_i, y_j, z_k)$, utilizing corrected differences at all interior mesh grids and taking correction terms as auxiliary variables, the PBE (5) can be discretized as:

$$L_h u_{i,j,k} + C_{i,j,k} - N(u_{i,j,k}) = 0, \tag{12}$$

where the nonlinear term $N(u_{i,j,k})$ is diagonal and $N(u_{i,j,k}) = \frac{\bar{\kappa}^2}{\epsilon^+}\sinh(u_{i,j,k})$ for each grid point in $\Omega^+$. The resulting equation (12) in matrix form is given by:

$$AU + BQ - N(U) = 0, \tag{13}$$

where $B$ is a sparse matrix of dimension $N_1$ by $3N_2$ consisting of coefficients from correction terms, and the symmetric and diagonally dominant matrix $A$ consists of coefficients from discretizing the 3D differential Laplacian operator obtained via the second-order central difference.

By combining Eq.(13) with Eq.(11), a nonlinear augmented system is constructed. We note that $N_2$ is much smaller than $N_1$. Thus, for computational efficiency, one usually first solves for $Q$ in a Schur complement form of the augmented system. In this study, a nonlinear relaxation scheme or inexact-Newton method is considered to address the nonlinearities inherent in the system, with detailed information provided in the next two subsections.

## 2.5. The AMIB-FFT algorithm

In a nonlinear relaxation scheme, the nonlinearity term of the PBE is evaluated by using function values of the previous step in an iterative loop. For example, the PBE (5) can be relaxed at the iterative step $n$ as:

$$\Delta u^n = \frac{\bar{\kappa}^2}{\epsilon^+} \sinh(u^{n-1}), \qquad r \in D \tag{14}$$

For the augmented system (13), the relaxation scheme is given by

$$AU^n + BQ^n = N(U^{n-1}), \tag{15}$$

in which matrix $A$ can be inverted by the FFT algorithm.

The AMIB-FFT method is implemented as a two-layer iterative algorithm within this augmented formulation. The process commences with the initialization of potentials, denoted as $U^0$, utilizing the solution to the Linear Poisson-Boltzmann equation (LPB) as an initial guess vector within the domain $D$. Additionally, $Q^0$ is set as $\Phi - CU^0$. The iteration steps involve two layers. In the initial layer, through the nonlinear relaxation scheme, the solution is updated in the outer iteration and is subsequently used in the next iteration. The outer iteration steps continue until the convergence condition $||U^n - U^{n-1}||_2 <$ outer tol is met. In each outer iteration, combining Eq. (15) with Eq. (11) at the $n^{th}$ step forms a new augmented system to be solved. A Schur complement system for $Q^n$ is solved using the Biconjugate Gradient (BCG) method, with the known $Q^{n-1}$ as the initial value:

$$(I - CA^{-1}B)Q^n = \Phi - CA^{-1}N(U^{n-1}). \tag{16}$$

The FFT inversion is performed on $BQ^n$ and $N(U^{n-1})$ in (16). Notably, the linear system (16) for $Q^n$ has a much smaller degree of freedom than $U$, specifically $N_2$. While $Q^n$ cannot be precisely obtained within a few iterations in (16), it is unnecessary to do so initially. Inspired by the concept of the Inexact-Newton method, the system is solved inexactly and $Q^n$ will be updated to start a new iteration until reaching $tol = 10^{-2}||U^{n-1} - U^{n-2}||_2$

without requiring numerous iteration steps in the early iterations. Subsequently, the solution is updated through the FFT inversion for

$$AU^n = N(U^{n-1}) - BQ^n.$$

The effectiveness of the second-order AMIB-FFT method relies on both the iteration count for the outer nonlinear relaxation and the inner biconjugate gradient iterations. It is worth noting that each inner iteration includes one FFT inversion with a complexity of $O(N_1 \log N_1)$. The iteration count for the biconjugate gradient method exhibits a weak dependence on the mesh size. However, the iteration count for the outer nonlinear relaxation is not guaranteed to be small.

### 2.6. The AMIB-Multigrid algorithm

The application of the inexact-Newton method to solve the AMIB discretization of the regularized nonlinear PBE involves a systematic three-layer iterative algorithm. To begin, the nonlinear algebraic system based on the AMIB discretization is represented as:

$$F(U,Q) = \begin{cases} AU + BQ - N(U) \\ CU + IQ - \Phi \end{cases} = 0 \tag{17}$$

The inexact-Newton method for the nonlinear system (17) can be formulated as:

$$F'(U^n, Q^n)(\delta U^n, \delta Q^n) = -\, F(U^n, Q^n), \tag{18}$$

$$\begin{bmatrix} U^{n+1} \\ Q^{n+1} \end{bmatrix} = \begin{bmatrix} U^n \\ Q^n \end{bmatrix} + \begin{bmatrix} \delta U^n \\ \delta Q^n \end{bmatrix} \tag{19}$$

The first layer iteration, denoted in (19), updates the solution using the correction terms $\delta U^n$ and $\delta Q^n$. $F'(U^n, Q^n)$ represents the Jacobian matrix, and by denoting $\tilde{A^n} = A - N'(U^n)$ in which $N'(U) = \frac{\bar{\kappa}^2}{\epsilon^+} \cosh(U)$, a new augmented system is formed:

$$\tilde{A^n} \delta U^n + B \delta Q^n = F_1^n \tag{20}$$

$$C \delta U^n + I \delta Q^n = F_2^n \tag{21}$$

Here, $F_1^n$ and $F_2^n$ are defined as $-(AU^n + BQ^n - N(U^n))$ and $-(CU^n + IQ^n - \Phi)$, respectively.

We compute $\delta Q^n$ using the Schur complement, effectively eliminating $\delta U^n$ from Eq. (20) and Eq. (21) to create a linear system for $\delta Q^n$:

$$(I - C\tilde{A^n}^{-1}B)\delta Q^n = F_2^n - C\tilde{A^n}^{-1}F_1^n \tag{22}$$

To solve for $Q^n$ in equation system (22), the GMRES could be utilized. Within each iteration, the geometric multigrid iteration is applied to solve $\tilde{A^n}^{-1}F_1^n$ and $\tilde{A^n}^{-1}B$. The GMRES iteration process continues until the relative error tolerance equaling $10^{-2}$ is reached. To obtain the solution $\delta U^n$, the Multigrid algorithm is utilized to solve:

$$\tilde{A^n}\delta U^n = F_1^n - B\delta Q^n \tag{23}$$

After determining $\delta Q^n$, we shift all known quantities to the right-hand side.

This AMIB-Multigrid method manifests as a three-layer iterative algorithm in this augmented formulation, where the correction term in the first layer is considered a rough solution for the subsequent second and third-layer iterations involving the GMRES method and the Multigrid algorithm, respectively.

## 3. Numerical comparison

In this section, we numerically compare three MIB schemes, i.e., rMIB, AMIB-FFT, and AMIB-Multigrid, for solving the nonlinear PB equation. Based on the same MIB interface treatments, these finite difference schemes are expected to be second order accurate. However, different iterative algorithms are employed in these three methods. The rMIB involves two-layer iterations, i.e., outer Newton iteration and inner BCG iteration for the potential $U$. The AMIB-FFT also involves two-layer iterations, i.e., outer relaxation process and inner iteration for solving the auxiliary variable $Q$. Note that the FFT inversion is not an iterative process. Because the dimension of $Q$ is much smaller than that of $U$, the inner iteration of the AMIB-FFT could be faster than that of the rMIB. However, the convergence rate of the outer Newton scheme could be much faster than that of the relaxation. Thus, it is unclear whether rMIB or AMB-FFT will be faster, without numerically testing them. For the AMIB-Multigrid method, three-layer iterations are needed, because the multigrid inversion is an iterative process. With the inner multigrid iteration, the GMRES is chosen for the middle iteration for $Q$, since the transpose operation is not easy to compute here. Even though the outer inexact-Newton iteration is fast, the AMIB-Multigrid scheme could be slower than the other two methods, due to the use of three layers. The summary comparing three MIB schemes is displayed in Table 1.

Table 1: Based on the same MIB interface treatment, the three MIB schemes employ different iterative algorithms for solving the nonlinear PB equation.

| **Scheme** | **Iterations** | | | **Molecular Surface** |
|---|---|---|---|---|
| | First Layer | Second Layer | Third Layer | |
| rMIB | Newton | BCG for $U$ | None | MSMS/ESES |
| AMIB-FFT | Relaxation | BCG for $Q$ | None | MSMS/ESES |
| AMIB-Multigrid | Newton | GMRES for $Q$ | Multigrid | MSMS/ESES |

### 3.1. A spherical cavity with analytical potential

We first consider a benchmark example with analytical potentials. The test involves a spherical cavity containing a single centered charge over the domain $\Omega = [-4, 4]^3$. The analytical potential and the artificial source term of the nonlinear PB equation are given as in [9].

$$\phi(r) = \begin{cases} \frac{1}{\epsilon R} - \frac{1}{R} + \frac{1}{|r|}, & |r| < R \\ \frac{1}{\epsilon |r|}, & |r| > R \end{cases} \tag{24}$$

$$\rho(r) = \begin{cases} 4\pi\delta(r), & |r| < R \\ \bar{\kappa}^2 \sinh(\frac{1}{\epsilon|r|}), & |r| > R \end{cases} \tag{25}$$

where $\epsilon = \epsilon^+/\epsilon^-$ and $R$ is the radius of the sphere. In this test, we take $\epsilon^+ = 80$, $\epsilon^- = 1$, and $R = 2\pi/3$. On $\partial\Omega$, a Dirichlet boundary condition obtained from the analytical potential (24) is imposed. Comparative results for the rMIB, AMIB-FFT, and AMIB-Multigrid methods are presented in Table 2. Here, the $L_\infty$ and $L_2$ errors of the potential near the interface are reported for different mesh sizes $n = n_x = n_y = n_z$. It can be seen from Table 2 that all three methods demonstrate the second order convergence pattern and the calculated potentials are very close. The efficiency of these methods is also reported in Table 2 through CPU time in seconds. With a tight domain $\Omega = [-4, 4]^3$, the AMIB-FFT scheme exhibits the highest efficiency among the three methods, while the AMIB-Multigrid method is comparatively slower due to its three layers of iteration.

Table 2: Numerical error in the potential solution and computational efficiency for the spherical cavity test.

| Scheme | $n$ | $h$ | $L_\infty$ |  | $L_2$ |  | CPU (s) |
|---|---|---|---|---|---|---|---|
|  |  |  | Error | Order | Error | Order |  |
| rMIB | 17 | 0.5 | 1.33E-4 |  | 3.46E-5 |  | 4.44E-2 |
|  | 33 | 0.25 | 3.21E-5 | 2.05 | 6.74E-6 | 2.36 | 5.12E-1 |
|  | 65 | 0.125 | 5.36E-6 | 2.58 | 9.18E-7 | 2.88 | 7.56E+0 |
|  | 129 | 0.0625 | 1.48E-6 | 1.86 | 2.12E-7 | 2.11 | 1.16E+2 |
|  | 257 | 0.03125 | 2.90E-7 | 2.35 | 4.42E-8 | 2.26 | 1.64E+3 |
| AMIB-FFT | 17 | 0.5 | 1.33E-4 |  | 3.46E-5 |  | 6.83E-2 |
|  | 33 | 0.25 | 3.21E-5 | 2.05 | 6.74E-6 | 2.36 | 5.55E-1 |
|  | 65 | 0.125 | 5.38E-6 | 2.58 | 9.65E-7 | 2.80 | 4.81E+0 |
|  | 129 | 0.0625 | 1.48E-6 | 1.86 | 2.17E-7 | 2.15 | 7.05E+1 |
|  | 257 | 0.03125 | 3.07E-7 | 2.27 | 4.67E-8 | 2.22 | 9.59E+2 |
| AMIB-Multigrid | 17 | 0.5 | 1.33E-4 |  | 3.46E-5 |  | 9.87E-1 |
|  | 33 | 0.25 | 3.21E-5 | 2.05 | 6.14E-6 | 2.49 | 5.00E+0 |
|  | 65 | 0.125 | 5.36E-6 | 2.58 | 9.18E-7 | 2.74 | 3.69E+1 |
|  | 129 | 0.0625 | 1.48E-6 | 1.86 | 2.12E-7 | 2.11 | 5.90E+2 |
|  | 257 | 0.03125 | 2.90E-7 | 2.35 | 4.42E-8 | 2.26 | 8.15E+3 |

### 3.2. Electrostatic free energy of the nonlinear Kirkwood sphere

We next consider a nonlinear Kirkwood sphere with radius $R = 2\pi/3$ in a finite computational domain $[-b, b]^3$ [3]. The only source is a point charge $q = 1$ at the center of sphere $[0, 0, 0]$, without using the artificial source in the solvent as in (25). Such a Kirkwood sphere does not admit an analytical potential, while we know physically that the potential $\phi$ will decay to zero exponentially as $|r| \to \infty$. In Ref. [3], the 3D nonlinear Kirkwood sphere problem is recasted into 1D boundary value problem (BVP) with a very large $b = 200$ and a Dirichlet zero boundary condition. The solvation free energy $E_{\text{sol}}$ can then be estimated by using an eighth order MIB (MIB8) scheme. For the present test, the reference energy is estimated to be $E_{r1} = -78.512760423667700$, with a numerical error around $10^{-12}$.

We first consider $b = 16$ for the 3D domain $\Omega = [-b, b]^3$ with a Dirichlet zero boundary condition on $\partial\Omega$. By using different $h$ values, the solvation free energies $E_{\text{sol}}$ calculated by three methods are listed in Table 3 and errors against the reference energy $E_{r1}$ are also reported. It can be seen that the calculated $E_{\text{sol}}$ values converge to the same place for the three schemes. However, at $h = 0.125$, the energy errors become larger and the numerical orders are negative. This is due to the physical setting. With $b = 16$, the Dirichlet zero boundary condition is a crude approximation, so that the solvation free energy $E_{\text{sol}}$ does not converge to the true energy defined over

Table 3: The nonlinear Kirkwood energy is calculated by the three schemes with a fixed $b = 16$ and different $h$ values. Here, the error is calculated against the reference energy $E_{r1} = -78.512760423667700$ generated by the MIB8 scheme [3].

| Scheme | $n$ | $h$ | $E_{\mathrm{sol}}$ | Error | Order | CPU (s) |
|---|---|---|---|---|---|---|
| rMIB | 33 | 1.0 | -78.45918823 | 5.36E-2 | – | 8.09E-1 |
| | 65 | 0.5 | -78.49758073 | 1.52E-2 | 1.82 | 1.18E+1 |
| | 129 | 0.25 | -78.51320211 | 4.42E-4 | 5.10 | 1.60E+2 |
| | 257 | 0.125 | -78.51683465 | 4.07E-3 | -3.21 | 2.48E+3 |
| AMIB-FFT | 33 | 1.0 | -78.45540886 | 5.74E-2 | – | 1.34E+0 |
| | 65 | 0.5 | -78.49707161 | 1.57E-2 | 1.87 | 1.46E+1 |
| | 129 | 0.25 | -78.51320212 | 4.42E-4 | 5.15 | 2.91E+2 |
| | 257 | 0.125 | -78.51683466 | 4.07E-3 | -3.20 | 2.00E+3 |
| AMIB-Multigrid | 33 | 1.0 | -79.30609914 | 7.93E-1 | – | 2.36E+0 |
| | 65 | 0.5 | -78.43701396 | 7.57E-2 | 3.39 | 3.57E+1 |
| | 129 | 0.25 | -78.51320211 | 4.42E-4 | 7.42 | 4.19E+2 |
| | 257 | 0.125 | -78.51683465 | 4.07E-3 | -3.20 | 3.33E+3 |

the infinite domain or $E_{r1}$ calculated based on $b = 200$. The CPU time of these three schemes are also shown in Table 3. The rMIB scheme is the fastest for coarse meshes, while the AMIB-FFT becomes faster than the rMIB when a dense mesh with $h = 0.125$ is used.

Table 4: The nonlinear Kirkwood energy is calculated by the three schemes with a fixed $h = 0.125$ and different $b$ values. Here, the error is calculated against the reference energy $E_{r1} = -78.512760423667700$ generated by the MIB8 scheme [3].

| Scheme | $b$ | $n$ | $E_{\mathrm{sol}}$ | Error | Order | CPU (s) |
|---|---|---|---|---|---|---|
| rMIB | 4 | 65 | -78.74814918 | 2.35E-1 | – | 1.36E+1 |
| | 8 | 129 | -78.56525601 | 5.25E-2 | 2.16 | 1.79E+2 |
| | 16 | 257 | -78.51683465 | 4.07E-3 | 3.69 | 2.48E+3 |
| | 32 | 513 | -78.51216799 | 5.92E-4 | 2.78 | 5.55E+4 |
| AMIB-FFT | 4 | 65 | -78.74814920 | 2.35E-1 | – | 2.38E+1 |
| | 8 | 129 | -78.56525605 | 5.25E-2 | 2.16 | 1.62E+2 |
| | 16 | 257 | -78.51683466 | 4.07E-3 | 3.69 | 2.00E+3 |
| | 32 | 513 | – | – | – | – |
| AMIB-Multigrid | 4 | 65 | -78.74814918 | 2.35E-1 | – | 4.89E+1 |
| | 8 | 129 | -78.56525601 | 5.25E-2 | 2.16 | 4.71E+2 |
| | 16 | 257 | -78.51683465 | 4.07E-3 | 3.69 | 3.33E+3 |
| | 32 | 513 | – | – | – | – |

The three schemes will produce a better energy if a larger $b$ value is used. To illustrate this point, we investigated the influence of the parameter $b$ on the energy error by fixing $h = 0.125$. The calculated energies and the corresponding errors against $E_{r1}$ are detailed in Table 4. Indeed, the energy error becomes smaller as $b$ becomes larger, and a quadratic convergence rate could be observed. However, when $b = 32$, both AMIB schemes fail to converge. The AMIB-Multigrid method fails due to an excessive runtime. For the AMIB-FFT scheme, its CPU time is the fastest for $n = 129$ and 257. The failure of the AMIB-FFT scheme at $n = 513$ or $b = 32$ is believed to be due to the instability of the relaxation scheme (15). Basically, with a fixed radius $R = 2\pi/3$, a larger $b$ means that more unknown values of $U$ are in the solvent domain $\Omega^+$, where the nonlinear term is evaluated. Thus, the nonlinear effect becomes more significant to the point that the relaxation scheme becomes unstable at $b = 32$. The rMIB scheme performs robustly in this tough nonlinear test.

Table 5: The nonlinear Kirkwood energy is calculated by the three schemes with fixed $b = 16$ and different $h$ values. Here, the error is calculated against the reference energy $E_{r2} = -78.5173353411318$ generated by the 3D rMIB scheme.

| Scheme | $n$ | $h$ | $E_{\text{sol}}$ | Error | Order | CPU (s) |
|---|---|---|---|---|---|---|
| rMIB | 33 | 1.0 | -78.45918823 | 5.81E-2 | – | 8.09E-1 |
| | 65 | 0.5 | -78.49758073 | 1.96E-2 | 1.56 | 1.18E+1 |
| | 129 | 0.25 | -78.51320211 | 4.13E-3 | 2.26 | 1.60E+2 |
| | 257 | 0.125 | -78.51683465 | 5.01E-4 | 3.04 | 2.48E+3 |
| AMIB-FFT | 33 | 1.0 | -78.45540886 | 6.19E-2 | – | 1.33E+0 |
| | 65 | 0.5 | -78.49707161 | 2.03E-2 | 1.61 | 1.46E+1 |
| | 129 | 0.25 | -78.51320212 | 4.13E-3 | 2.29 | 2.91E+2 |
| | 257 | 0.125 | -78.51683466 | 5.01E-4 | 3.04 | 2.00E+3 |
| AMIB-Multigrid | 33 | 1.0 | -79.30609914 | 7.89E-1 | – | 2.36E+0 |
| | 65 | 0.5 | -78.43701396 | 8.03E-2 | 3.29 | 3.57E+1 |
| | 129 | 0.25 | -78.51320211 | 4.13E-3 | 4.28 | 4.19E+2 |
| | 257 | 0.125 | -78.51683465 | 5.01E-4 | 3.04 | 3.33E+3 |

The non-convergence issue underlying Table 3 is due to an insufficiently large value of $b$, so all three schemes cannot converge to the true energy. To further examine the numerical convergence, we restricted again to the domain size $b = 16$ with the Dirichlet zero boundary condition. Instead of using the 1D MIB8 code [3], a different reference energy is generated by the 3D rMIB scheme with $h = 0.0625$ or $n = 513$, which is denoted as $E_{r2} = -78.5173353411318$. By benchmarking with $E_{r2}$, the energy errors of three schemes are reported in Table 5. It can be observed that the energies for all three methods converge to $E_{r2}$, and the numerical orders are essentially two.

### 3.3. Electrostatics on Proteins

We proceeded by solving the nonlinear PB equation for several proteins to assess the accuracy and efficiency of the three methods in handling complex geometries and charge distributions. For protein simulations, the MSMS molecular surface [21] was assumed in the original rMIB [10], while the ESES molecular surface has recently been incorporated in the rMIB package [2]. The solvation energies of selected proteins, computed by the three methods at different mesh sizes, are reported in Table 6 and Table 7, employing MSMS and ESES, respectively. The Dirichlet boundary condition (2) is assumed in all protein cases. With a glimpse, we could see that the solvation energies from the three methods are almost consistent. However, a reduction in efficiency is observed for both AMIB methods, attributable to three factors.

Firstly, the rMIB computation can be carried out over a tight domain $\Omega$, while a larger immerse domain $D = \Omega \cup \Omega^e$ has to be employed in the AMIB methods, so that the grid numbers for $D$ are powers of 2. In both tables, grid numbers $(n_x, n_y, n_z)$ of the rMIB domain $\Omega$ are reported. Some special $h$ values are selected for these proteins such that $n_x$, $n_y$, and $n_z$ are smaller but quite close to some powers of 2. In these cases, the extra grid nodes introduced in $\Omega^e$ are not too many. However, when a common mesh size $h = 0.5$ is employed, a significant portion of the CPU time could be wasted for the artificial domain $\Omega^e$. Secondly, the augmented formulation involves more auxiliary variables in $Q$ as the geometric interface becomes more intricate. Finally, the AMIB-FFT method is faster than the AMIB-Multigrid scheme in most cases, because the AMIB-Multigrid scheme requires three layers of iteration.

## 4. Conclusion

In summary, our investigation involved a comprehensive comparison of three MIB schemes – rMIB, AMIB-FFT, and AMIB-Multigrid – within the regularization framework for solving the nonlinear PBE. Based on the same MIB interface treatment and central difference approximation, all three schemes produce almost the same accuracy in all tested cases and exhibit the second order of convergence. The AMIB-Multigrid scheme is less efficient than the

Table 6: Solving the nonlinear PB equation for proteins with the MSMS molecular surface.

| PDB | $h$ | $[n_x, n_y, n_z]$ | rMIB | | AMIB-FFT | | AMIB-Multigrid | |
|---|---|---|---|---|---|---|---|---|
| | | | $E_{\mathrm{sol}}$ | CPU (s) | $E_{\mathrm{sol}}$ | CPU (s) | $E_{\mathrm{sol}}$ | CPU (s) |
| 2erl | 0.7 | $[45, 43, 57]$ | -957.58 | 3.47E+0 | -957.41 | 1.08E+1 | -957.40 | 6.61E+1 |
| | 0.5 | $[63, 61, 79]$ | -952.54 | 1.45E+1 | -952.43 | 2.03E+1 | -952.46 | 1.79E+2 |
| | 0.35 | $[89, 87, 113]$ | -953.02 | 4.20E+1 | -952.86 | 8.69E+1 | -952.89 | 6.67E+2 |
| 1cnb | 1 | $[37, 33, 39]$ | -307.77 | 1.26E+0 | -307.61 | 9.81E+0 | -307.59 | 2.76E+1 |
| | 0.7 | $[53, 47, 53]$ | -306.80 | 4.11E+0 | -306.74 | 1.96E+1 | -306.74 | 6.56E+1 |
| | 0.5 | $[73, 65, 75]$ | -303.38 | 1.68E+1 | -303.37 | 8.26E+1 | -303.36 | 2.79E+2 |
| | 0.3 | $[121, 107, 125]$ | -303.94 | 7.60E+1 | -303.92 | 1.44E+2 | -303.96 | 6.66E+2 |
| 1ajj | 0.7 | $[45, 49, 55]$ | -1214.38 | 4.18E+0 | -1213.79 | 1.01E+1 | -1213.56 | 7.09E+1 |
| | 0.5 | $[63, 67, 77]$ | -1201.72 | 1.53E+1 | -1201.64 | 9.13E+1 | -1201.60 | 4.07E+2 |
| | 0.35 | $[89, 95, 109]$ | -1203.80 | 4.70E+1 | -1203.55 | 3.18E+2 | -1203.43 | 1.13E+3 |
| 1a63 | 0.8 | $[87, 55, 57]$ | -2420.45 | 1.10E+1 | -2420.27 | 2.49E+2 | -2420.20 | 1.82E+2 |
| | 0.5 | $[137, 87, 89]$ | -2382.25 | 6.86E+1 | -2381.92 | 1.42E+3 | -2381.87 | 1.49E+3 |

Table 7: Solving the nonlinear PB equation for proteins with the ESES molecular surface.

| PDB | $h$ | $[n_x, n_y, n_z]$ | rMIB | | AMIB-FFT | | AMIB-Multigrid | |
|---|---|---|---|---|---|---|---|---|
| | | | $E_{\mathrm{sol}}$ | CPU (s) | $E_{\mathrm{sol}}$ | CPU (s) | $E_{\mathrm{sol}}$ | CPU (s) |
| 1ajj | 0.7 | $[42, 45, 49]$ | -1531.44 | 2.90E+0 | -1531.19 | 1.13E+1 | -1530.44 | 7.43E+1 |
| | 0.3 | $[97, 105, 115]$ | -2122.51 | 6.51E+1 | -2120.36 | 1.04E+2 | -2120.53 | 7.21E+2 |
| 1a63 | 1.1 | $[61, 39, 39]$ | -2415.10 | 4.13E+0 | -2418.03 | 7.24E+1 | -2416.24 | 1.61E+2 |

other two schemes because it involves three layers of iterations in solving the nonlinear augmented system. The AMIB-FFT scheme could be faster than the rMIB method for spherical cavity and Kirkwood sphere studies. However, due to the use of an extended domain with extra unknowns, the AMIB-FFT scheme can not compete with the rMIB scheme in terms of efficiency for all protein studies. Moreover, the nonlinear relaxation of the AMIB-FFT could experience instability issues, when there are a lot of unknowns in the solvent domain, for which the nonlinear term needs to be evaluated at the previous iterative step. In conclusion, the present comparison indicates that the rMIB scheme is the best method among the three MIB schemes for solving the nonlinear PB equation, in terms of accuracy, efficiency, and robustness.

Our study provided a comprehensive understanding of the strengths and limitations of the rMIB, AMIB, relaxation, Newton, FFT, and multigrid methods in solving the nonlinear PB equation. These investigations will benefit our further development of an optimal approach for solving the PB equation based on specific computational requirements and accuracy considerations. For the nonlinear Kirkwood sphere problem, our results demonstrate a strong dependence of accuracy on the domain size $b$, when a Dirichlet boundary condition is used. If the usual Dirichlet boundary condition (2) is used, a similar dependence could be observed and will be reported elsewhere. A more serious issue behind that is the lack of a physical boundary condition for the nonlinear PB model, and a solution to this issue is under our investigation.

## Acknowledgments

Zhao's research is partially supported by the National Science Foundation (NSF) of USA under grants DMS-2110914 and DMS-2306991. Geng's research is partially supported by the NSF under grant DMS-2110922.

## References

[1] S. Ahmed Ullah and S. Zhao, Pseudo-transient ghost fluid methods for the Poisson-Boltzmann equation with a two-component regularization. *Applied Mathematics and Computation*, **380**, 125267, 2020.

[2] S. Ahmed Ullah, X. Yang, B. Jones, S. Zhao, W. Geng, and G.W. Wei, Bridging Eulerian and Lagrangian Poisson-Boltzmann solvers by ESES. *Journal of Computational Chemistry*, **45**, 306-320, 2024.

[3] S. Amihere, W. Geng, and S. Zhao, Benchmarking electrostatic free energy of the nonlinear Poisson-Boltzmann model for the Kirkwood sphere. *Communications in Information and Systems*, **22**(3), 305-315, 2022.

[4] D. Chen, Z. Chen, C. Chen, W. Geng, and G.W. Wei, MIBPB: a software package for electrostatic analysis. *Journal of Computational Chemistry*, **32**, 657-670, 2011.

[5] H. Feng and S. Zhao, FFT-based high order central difference schemes for the three-dimensional Poisson equation with various types of boundary conditions. *Journal of Computational Physics*, **410**, 109391, 2020.

[6] H. Feng and S. Zhao, A fourth order finite difference method for solving elliptic interface problems with the FFT acceleration. *Journal of Computational Physics*, **419**, 109677, 2020.

[7] F. Fogolari, A. Brigo, H. Molinari, The Poisson-Boltzmann equation for biomolecular electrostatics: a tool for structural biology. *J. Mol. Recognit.*, **15**(6), 377–392, 2002.

[8] W. Geng, S. Yu, and G.W. Wei, Treatment of charge singularities in implicit solvent models. *Journal of Chemical Physics*, **127**, 114106, 2007.

[9] W. Geng, S. Zhao, Fully implicit ADI schemes for solving nonlinear Poisson–Boltzmann equation. *Mol. Based Math. Biol.*, **1**, 109–123, 2013.

[10] W. Geng and S. Zhao, A two-component Matched Interface and Boundary (MIB) regularization for charge singularity in implicit solvation. *Journal of Computational Physics*, **351**, 25-39, 2017.

[11] Michael J. Holst, The Poisson-Boltzmann Equation: Analysis and Multilevel Numerical Solution. PhD thesis, UIUC, 1994.

[12] C. Li, Y. Ren, G. Long, E. Boerman, and S. Zhao, A fast Sine transform accelerated high order finite difference method for parabolic problems over irregular domains. *Journal of Scientific Computing*, **95**, 49, 2023.

[13] Liu, Beibei, Bao Wang, Rundong Zhao, Yiying Tong, and Guo Wei Wei, Eses: Software for Eulerian solvent excluded surface. 2017, 446-466.

[14] B.Z. Lu, Y.C. Zhou, M.J. Holst, J.A. McCammon, Recent progress in numerical methods for the Poisson-Boltzmann equation in biophysical applications. *Commun. Comput. Phys.*, **3**(5), 973–1009, 2008.

[15] Luty, Brock A., Malcolm E. Davis, and J. Andrew McCammon, Solving the finite difference nonlinear Poisson Boltzmann equation. *Journal of computational chemistry*, **13**(9), 1114-1118, 1992.

[16] HOLST MJ, SAIED F., Numerical solution of the nonlinear Poisson-Boltzmann equation: developing more robust and efficient methods. *Journal of Computational Chemistry*, **16**(3), 337–364, 1995.

[17] Nicholls, Anthony, and Barry Honig, A rapid finite difference algorithm, utilizing successive over-relaxation to solve the Poisson-Boltzmann equation. *Journal of computational chemistry*, **12**(4), 435-445, 1991.

[18] Oberoi, Himanshu, and Norma M. Allewell, Multigrid solution of the nonlinear Poisson-Boltzmann equation and calculation of titration curves. *Biophysical journal*, **65**(1), 48-55, 1993.

[19] W. Rocchia, E. Alexov, and B. Honig, Extending the applicability of the nonlinear Poisson-Boltzmann equation: Multiple dielectric constants and multivalent ions (vol 105B, pg 6507, 2001). *Journal of Physical Chemistry B*, **105**(28), 6754-6754, 2001.

[20] Y. Ren and S. Zhao, A FFT accelerated fourth order finite difference method for solving three-dimensional elliptic interface problems. *Journal of Computational Physics*, **477**, 111924, 2023.

[21] M.F. Sanner, A.J. Olson, J.C. Spehner, Reduced surface: an efficient way to compute molecular surfaces. *Biopolymers*, **38**, 305–320, 1996.

[22] S. Yu, W. Geng, and G.W. Wei, Treatment of geometric singularities in implicit solvent models. *Journal of Chemical Physics*, **126**, 244108, 2007.

[23] S. Zhao and G. W. Wei, Matched interface and boundary (MIB) for the implementation of boundary conditions in high order central finite differences. *Int. J. Numer. Methods Eng.*, **77**, 1690-1730, 2009.

[24] Y.C. Zhou, S. Zhao, M. Feig, and G. W. Wei, High order matched interface and boundary method for elliptic equations with discontinuous coefficients and singular sources. *Journal of Computational Physics*, **213**, 1-30, 2006.

Yiming Ren
Department of Mathematics, University of Alabama
Tuscaloosa, AL 35487, USA
*E-mail address:* yren24@crimson.ua.edu

Sylvia Amihere
Department of Mathematics, University of Alabama
Tuscaloosa, AL 35487, USA
*E-mail address:* samihere@crimson.ua.edu

Weihua Geng
Department of Mathematics, Southern Methodist University
Dallas, TX 75275 USA
*E-mail address:* wgeng@mail.smu.edu

Shan Zhao
Department of Mathematics, University of Alabama
Tuscaloosa, AL 35487, USA
*E-mail address:* szhao@ua.edu

Received 23 January 2024