Discontinuous Bubble Immersed Finite Element Method for Poisson-Boltzmann Equation

In Kwon and Do Y. Kwak

1 Department of Mathematical Science, KAIST, 291 Daehak-ro (373-1 Guseong-dong), Yuseong-gu, Daejeon 305-701, Republic of Korea.

Received 16 January 2018; Accepted (in revised version) 9 May 2018

Abstract. We develop a numerical scheme for nonlinear Poisson-Boltzmann equation. First, we regularize the solution of PBE to remove the singularity. We introduce the discontinuous bubble function to treat the nonhomogeneous jump conditions of the regularized solution. Next, starting with an initial guess, we apply linearization to treat the nonlinearity. Then, we discretize the discontinuous bubble and the bilinear form of PBE. Finally, we solve the discretized linear problem by IFEM. This process is repeated by updating the previous approximation.

We carry out numerical experiments. We observe optimal convergence rate for all examples.

AMS subject classifications: 65F10, 65N30, 65Z05, 92C40

Key words: Biomolecular electrostatics, Poisson-Boltzmann equation, immersed finite element method, discontinuous bubble function, linearization.

1 Introduction

Poisson-Boltzmann (PB) theory is a well-established theory in various disciplinary areas. It is known as Gouy-Chapman theory in electrochemistry [3, 12], as Debye-Huckel theory in solution chemistry [9], as the Derjaguin-Landau-Verwey-Overbeek (DLVO) theory in colloid chemistry [10, 36] and as PB theory in biophysics [8, 17]. The Poisson-Boltzmann equation (PBE) is applied mainly as a modeling tool to make approximations for applications such as charged biomolecular interactions, dynamics of electrons in semiconductors or plasma. For example, PBE can be used to obtain the electrostatic potential and free energy of charged molecule like tRNA in an ionic solution with diverse number of bound ions.

*Corresponding author. Email addresses: kiki3568@kaist.ac.kr (I. Kwon), kdy@kaist.ac.kr (D. Y. Kwak)
The fundamental physics and theories of implicit solvent models are discussed in [33, 34]. The PBE represents a typical implicit solvent model, and gives a simplified continuum description of the discrete particle (e.g., water, molecule) distributions in solution. The PBE expresses the electrostatic interaction and ionic density distributions in a solvated system at the equilibrium state. The equation has singular charges at discrete particles. It makes a solution singular and interrupts general approaches to solve the equation. Also, solvated biomolecular systems are usually modeled by dielectrically distinct areas with singular charges distributed in the molecular areas. The PBE has dielectric coefficient dependent on these distinct areas. It results in the discontinuity of solution on each areas.

Many numerical methods were developed for the Poisson-Boltzmann equation. Finite difference methods (FDM) have been the most popular numerical methods for the PBE in biomolecular simulations, most likely due to their simpler implementation than other methods. DelPhi [20], GRASP [32], MEAD [1], UHBD [31] are the most popular and successful traditional FDM based solver for obtaining biomolecular electrostatics. Since several features of the PBE were untreated in these traditional PBE solver, there is a loss of accuracy and convergence rate [39]. These untreated features are the approximated position of molecular surface and the continuity condition of electric displacement on the molecular surface. Later, Jump Condition capturing scheme (JCCS) [37] for structured meshes, Matched Interface and Boundary method (MIB) [38,40,41,44], and Newton algebraic multigrid (AMG) [13, 15, 16] for nonlinear PBE have been developed [45].

On other hand, Finite Element methods (FEM) have been developed for Poisson type of PDE in many different ways. Compared with the FDM, FEM provide more flexibility for fitting curved domain and local refinement, and more rigorous convergence analysis. Also more choices of efficient iterative solvers for the resultant linear systems and more schemes for solving nonlinear equations are allowed. Because the PBE has singular partial charges, rigorous theory for the solution and approximation theory were not well-developed until recently. In [4], first rigorous a priori error estimates and rigorous convergence results were established using a decomposition scheme. This article provides a number of fundamental technical results. Also, Newton-AMG iterations [14–16]
and Newton-Krylov iterations [35] have been developed to handle nonlinearity of PBE.

So far, most of popular schemes for interface problem were one based on fitted grids. Since the schemes based on fitted grids have an unstructured grids, they result in a discretized system with complicated data structure. We introduce a new scheme for the PBE with interface conditions, based on structured grids. Immersed finite element methods (IFEM) use structured grid to handle the interface problem. IFEM was introduced by Z. Li, T. Lin and Y. Lin and their coworkers [26, 27] and later developed by Kwak et al., to side class of application [6, 24]. The main advantage of IFEM is that IFEM use structured grids so that the data structure of discretized system is simpler than those of fitted grids and it is easy to develop efficient solvers. The convergence analysis for the early version of IFEM is given in [6, 23, 24] for the homogeneous jump cases. IFEM has been applied to other problems such as multiphase flows in porous media [19], Robin type elliptic problems [21] and elasticity problem [18, 22, 25]. IFEM based on Crouzeix-Raviart (CR) $P_1$-nonconforming space [7] was introduced in [24]. Also, IFEM was enriched by consistency terms of DG [23, 28]. Since it can use uniform grids, we can apply the geometric multigrid solver. The performance of geometric multigrid was first reported in [19].

To handle the problem with nonhomogeneous jumps across the interface, the discontinuous bubble IFEM (DB-IFEM) [2] was introduced. In DB-IFEM, they construct a discontinuous bubble type function which satisfies the given jump conditions. They obtain a new equation with homogeneous jump conditions by subtracting the discontinuous bubble function. Then, they solve the homogeneous problems by using schemes developed earlier [2, 24]. Since DB-IFEM has smaller number of degrees of freedom (dof) compared with DG, it has much smaller system to solve.

In this work, we introduce a method to solve PBE numerically based on IFEM. To apply IFEM, we need to regularize the PBE [5]. Since the governing equation of PBE has singularities, we decompose the solution into a regular part and singular part. Then we obtain a new equation for regular solution by certain regularization process. As a result of the regularization, the new governing equation for a regular solution has nonhomogeneous jumps in the solution and the flux. To treat these nonhomogeneous jumps we introduce a bubble function. Then we obtain a modified equation with homogeneous jumps. Next, we linearize the equation using Taylor expansion. We apply DB-IFEM to the linearized equation, and construct explicit form of discretized DB function. Thus, we obtain a discrete solution of PBE by applying DB-IFEM. DB-IFEM is simple to implement and treat the nonhomogeneous jumps efficiently compared with other schemes. Finally, We tested our scheme to some sample examples. For all examples we tested, we observe the optimal convergence rates, $O(h^2)$ in $L^2$ and $O(h)$ in $H^1$.

We compare the numerical performance with the existing methods mentioned above. As for the accuracy, our scheme shows perfect order in $H^1$ and $L^2$ norm, which is superior or at least as accurate as all softwares listed above. For the performance of the solver, our nonlinear Newton method converges in 4 iterations which is reasonably fast, while the inner linear problem is solved by the usual conjugate gradient method. Hence, we believe our scheme performs numerically as good as the other softwares.
Even though we designed the scheme for 2-D problems, it can be generalized to 3-D problems. To solve a 3D problem, we first construct a 3-D bubble function. Once a bubble function is constructed, we can proceed almost in the same way as the 2-D case. The details are the subjects of the next paper.

The rest of the paper is organized as follows. The governing equation of PBE and regularization of solution of PBE are described in Section 2. In Section 3, we introduce a DB function for nonhomogeneous jumps and Newton iteration to handle nonlinearity. In Section 4, we describe DB-IFEM based on P1-conforming. The numerical experiments are given in Section 5. The conclusion follows in Section 6.

2 Preliminary

2.1 Nonlinear Poisson-Boltzmann equation

We consider the PBE on the domain $\Omega = \Omega^+ \cup \Omega^- \subset \mathbb{R}^2$ where $\Omega^+$ is the solvent region with dielectric constant $\epsilon^+ > 0$ and $\Omega^-$ is the molecular region with dielectric constant $\epsilon^- > 0$. The molecular surface, denoted by $\Gamma$, is assumed to be $C^2$-continuous and $\partial \Omega$ denotes the boundary of the entire domain. The governing equation is following:

$$- \nabla \cdot (\epsilon \nabla u) + \kappa^2 \sinh(u) = \rho^f, \quad \rho^f = \sum_{i=1}^{N} q_i \delta(x-x_i), \quad (2.1a)$$

$$[u]_{\Gamma} = 0 \quad \text{on} \ \Gamma, \quad (2.1b)$$

$$\left[ \epsilon \frac{\partial u}{\partial n} \right]_{\Gamma} = 0 \quad \text{on} \ \Gamma, \quad (2.1c)$$

$$u = 0 \quad \text{on} \ \partial \Omega, \quad (2.1d)$$

where $u$ is an electrostatic potential, $\rho^f$ is the singular charge distribution, and $\kappa$ is the related parameter in $\Omega^+$, and is equal to 0 in $\Omega^-$. In Eq. (2.1a), $x_i$ is the location of charge, $q_i$ is amount of charge, and $\delta(x)$ is the Dirac delta function. $n_{\Gamma}$ is a unit normal vector to $\Gamma$ and bracket $[\cdot]_{\Gamma}$ denotes the jump across the interface, i.e., $[u]_{\Gamma} = u|_{\Omega^-} - u|_{\Omega^+}$.

2.2 Regularization of PBE

We give the definition of some function spaces and their norms. For any bounded sub-domain $D \subset \Omega$, we denote $D^+ = D \cap \Omega^+$, $D^- = D \cap \Omega^-$. We define $H^m(D), H^m_0(D), H^m(\partial D)$ to be the ordinary Sobolev spaces of order $m$ with the norm $\| \cdot \|_{m,D}$ and the semi-norm $| \cdot |_{m,D}$. For $m = 1,2$, the space $\tilde{H}^m (D)$ is defined as

$$\tilde{H}^m (D) := H^m (D^+) \cap H^m (D^-),$$

with norms

$$\| u \|^2_{\tilde{H}^m (D)} := \| u \|^2_{H^m (D^+)} + \| u \|^2_{H^m (D^-)}.$$
The subspace $\tilde{H}^1_0(D)$ is defined as
\[ \tilde{H}^1_0(D) := \{ u \in \tilde{H}^1(D) \mid u = 0 \text{ on } \partial D \}. \]

Since PBE has a singularity, it is not easy to solve by conventional numerical methods. The singular charge distribution of the PBE makes its solution discontinuous. It is well known [4] that the solution of PBE does not belong to $H^1(\Omega)$. Hence the standard FEM cannot be applied. So we shall express the original solution $u$ as sum of a regular solution $u^r$ and a singular potential $u^s$ which takes care of singularities [5].

\[ u = u^r + u^s. \] (2.2)

Then, the original equation is converted into another equation for the regular solution $u^r$. There are many choices of $u^s$. We choose it as follows:

\[ u^s = G|_{\Omega^-} \text{ in } \Omega^-, \]
\[ u^s = 0 \text{ in } \Omega^+, \]

where $G$ is defined by $G = -\sum_{i=1}^{N} \frac{q_i}{2\pi |x-x_i|}$ in $\mathbb{R}^2$ satisfying $-\epsilon^{-} \Delta G = \rho^f$. Using this choice of $u^s$ and (2.2), we obtain the following equation for the regular potential $u^r$ from (2.1).

\[ -\nabla \cdot (\epsilon \nabla u^r) + \kappa^2 \sinh(u^r) = 0 \text{ in } \Omega, \] (2.3a)
\[ [u^r] = f_1 \text{ on } \Gamma, \] (2.3b)
\[ \left[ \frac{\epsilon}{\partial n} \frac{\partial u^r}{\partial n} \right] = f_2 \text{ on } \Gamma, \] (2.3c)
\[ u^r = 0 \text{ on } \partial \Omega, \] (2.3d)

where $f_1 = -u^s$ and $f_2 = -\epsilon^{-} \frac{\partial u^s}{\partial n}$. Here, we have exploited the fact that $\kappa$ is equal to 0 in $\Omega^-$ and $u^s$ vanishes in $\Omega^+$. Since the solution of (2.3) satisfies jump conditions, we need to define affine spaces where jump conditions are imposed. We define $U_{f_1,f_2}(\Omega)$ and $U_0(\Omega)$ as follows:

\[ U_{f_1,f_2}(\Omega) := \{ u \in \tilde{H}^2(\Omega) \mid [u] = f_1 \text{ and } \left[ \frac{\epsilon}{\partial n} \frac{\partial u}{\partial n} \right] = f_2 \text{ on } \Gamma \}, \]
\[ U_0(\Omega) := \{ u \in U_{f_1,f_2}(\Omega) \mid u|_{\partial \Omega} = 0 \}. \]

As the result of this regularization, the singularity of the original equation has been transformed into interface jumps of a new equation.

There are a few numerical methods for PBE using different regularization schemes. In methods introduced in [4], a small relative error in the numerical solution of regular component appears very large in the full potential since the singular component is defined in the entire $\Omega$. In methods introduced in [11], they introduced an additional harmonic
function to treat the interface jump of the solution and used another scheme to treat the nonhomogeneous jump in the flux. JCCS which treats the interface jumps in [5] has less flexibility for local mesh refinement especially along the interface. Compared with these methods, the discontinuous bubble IFEM introduced in our work is simple to implement and treat the jumps in the solution and the flux at once. Also, the bubble function has a small support and is not affected on the shape of the interface. In the next section, we describe the scheme to treat the nonhomogeneous conditions.

3 DB for nonhomogeneous conditions and iteration by linearization

3.1 Weak formulation with DB

We will apply DB scheme introduced in [2]. We multiply \( v \in H^1_0(\Omega) \) on each side of (2.3a) and apply Green’s theorem to each subdomain \( \Omega^+ \) and \( \Omega^- \) to obtain

\[- \int_{\partial \Omega} \epsilon \frac{\partial u'}{\partial n} v ds + \int_{\Omega} \epsilon \nabla u' \cdot \nabla v dx + \int_{\Omega} \kappa^2 \sinh(u') v dx = 0, \quad \forall v \in H^1_0(\Omega),\]

where \( n \) is the unit outward normal vector to \( \Omega^i \) \( (i = +, -) \). Adding two equations for \( i = +, - \) and using (2.3c), we obtain the following:

\[ \sum_{i = +, -} \left( \int_{\Omega} \epsilon \nabla u' \cdot \nabla v dx + \int_{\Omega} \kappa^2 \sinh(u') v dx \right) = \int_{\Gamma} J_2 v ds, \quad \forall v \in H^1_0(\Omega). \quad (3.1) \]

Hence we obtain the weak form of (2.3) as follows: Find \( u \in \widetilde{H}^2(\Omega) \) satisfying (3.1), (2.3b) and (2.3d). Since the solution of the equation belongs to the affine space \( U_{\Omega, j}(\Omega) \), we need to subtract the nonhomogeneous part to work on a linear space. Among many possible choices of such functions, we recommend the one having support on a thin strip, called \( S_{\Gamma} \), containing \( \Gamma \) (see Fig. 2). We define a bubble function \( u^* \) in \( \widetilde{H}^2(S_{\Gamma}) \cap \widetilde{H}^1_0(S_{\Gamma}) \) satisfying

\[ [u^*] = J_1, \quad \left[ \epsilon \frac{\partial u^*}{\partial n} \right]_{\Gamma} = f_2. \]

We let \( \tilde{u} = u' - u^* \). Then we easily see that \( \tilde{u} \in H^1_0(\Omega) \). Also, we get the equation for \( \tilde{u} \):

\[- \nabla \cdot (\epsilon \nabla \tilde{u}) + \kappa^2 \sinh(\tilde{u} + u^*) = \nabla \cdot (\epsilon \nabla u^*) \quad \text{in } \Omega \setminus \Gamma, \quad (3.2a)\]
There are two approaches to this kind of nonlinear equation. The first method is to apply linearization to (3.2) and find a weak form. The second method is to write (3.2) as a weak form and then apply linearization to it. We will follow the second approach. Hence, the weak form for \( \bar{u} \) is following: Find \( \bar{u} \in H^1_0(\Omega) \) satisfying

\[
\sum_{i=+,-} \left( \int_{\Omega_i} e \nabla \bar{u} \cdot \nabla v \, dx + \int_{\Omega_i} \kappa^2 \sinh(\bar{u} + u^*) v \, dx \right) = \int_{\Gamma} J_2 v \, ds - \sum_{i=+,-} \int_{\Omega_i} e \nabla u^* \cdot \nabla v \, dx, \quad \forall v \in H^1_0(\Omega). \tag{3.3}
\]

Since \( \sinh(\bar{u} + u^*) \) is a nonlinear function, the left side of (3.3) cannot be defined as a bilinear form of \( \bar{u} \) and \( v \). In the next subsection, we introduce a linearized form to treat this difficulty.

### 3.2 Iteration by linearization

We use iterations to solve (3.3) with \( u^* \) fixed. Let \( \bar{u} = u^n + \delta u \) \((i = 0,1,2,\cdots)\), where \( u^n \) is \( n \)-th iteration. We consider the following Taylor expansion.

\[
\sinh(\bar{u} + u^*) = \sinh(u^n + \delta u + u^*) \\
\approx \sinh(u^n + u^*) + \cosh(u^n + u^*) \delta u. \tag{3.4}
\]
Finally, we rewrite the (3.5) as follows: Find \( \delta u \) satisfying

\[
- \nabla \cdot \epsilon (\nabla \delta u) + \kappa^2 \cosh(u^n + u^s) \delta u \\
= \nabla \cdot \epsilon \nabla (u^n + u^s) - \kappa^2 \sinh(u^n + u^s) \quad \text{in } \Omega \setminus \Gamma,
\]

(3.5a)

\[
[\delta u]_{\Gamma} = 0 \quad \text{on } \Gamma,
\]

(3.5b)

\[
\left[ \frac{\partial \delta u}{\partial n} \right]_{\Gamma} = 0 \quad \text{on } \Gamma,
\]

(3.5c)

\[
\delta u = 0 \quad \text{on } \partial \Omega.
\]

(3.5d)

Hence, new variational form is obtained as follows:

\[
\sum_{i=+, -} \left( \int_{\Omega^i} e \nabla \delta u \cdot \nabla v d\Omega + \int_{\Omega^i} \kappa^2 \cosh(u^n + u^s) \delta uv d\Omega \right)
\]

\[
= \int_{\Omega} e \nabla u^s \cdot \nabla v d\Omega - \kappa^2 \sinh(u^n + u^s) \delta v d\Omega \quad \forall \delta v \in H^1_0(\Omega). \quad (3.6)
\]

Here, we introduce the bilinear form \( a(\cdot, \cdot) \) for \( \delta u \) and \( v \) as follows:

\[
a(\delta u, v) := \sum_{i=+, -} \left( \int_{\Omega^i} e \nabla \delta u \cdot \nabla v d\Omega + \int_{\Omega^i} \kappa^2 \cosh(u^n + u^s) \delta uv d\Omega \right),
\]

\[
\forall v \in H^1(\Omega^+) \cap H^1(\Omega^-).
\]

Finally, we rewrite the (3.5) as follows: Find \( \delta u \in H^1_0(\Omega) \) satisfying

\[
a(\delta u, v) = <J_2, v>_{\Gamma} - e(\nabla (u^n + u^s), \nabla v) - \kappa^2 (\sinh(u^n + u^s), v), \quad \forall v \in H^1_0(\Omega), \quad (3.7)
\]

where \( \langle \cdot, \cdot \rangle \) denotes the \( L^2(\Omega) \) inner product and \( <\cdot, \cdot>_{\Gamma} \) denotes the \( L^2(\Gamma) \) inner product. We shall approximate \( \bar{u} \) by finding \( \delta u \) satisfying (3.7) and adding it to \( u^n \) iteratively.

**Lemma 3.1.** The weak problem (3.7) is equivalent to (3.5).

**Proof.** It suffices to show that (3.7) implies (3.5). Suppose that (3.7) holds. Restricting to \( \delta u \in \Omega^+ \cap H^1_0(\Omega^-) \), we obtain:

\[
\int_{\Omega^+} e \nabla \delta u \cdot \nabla v d\Omega + \int_{\Omega^-} \kappa^2 \cosh(u^n + u^s) \delta uv d\Omega
\]

\[
= - \int_{\Omega^+} e \nabla (u^n + u^s) \cdot \nabla v d\Omega - \int_{\Omega^-} \kappa^2 \sinh(u^n + u^s) v d\Omega \quad \text{for } s = + \text{ or } -.
\]

Hence, the following equations hold on each subdomain \( \Omega^+ \) and \( \Omega^- \) respectively.

\[
- \nabla \cdot \epsilon (\nabla \delta u) + \kappa^2 \cosh(u^n + u^s) \delta u \\
= \nabla \cdot \epsilon \nabla (u^n + u^s) - \kappa^2 \sinh(u^n + u^s) \quad \text{a.e. on } \Omega \setminus \Gamma.
\]

(3.8)
This gives (3.5a). Next, the left side of (3.7) becomes as follows by integrating by parts:

\[
\int_{\partial \Omega^+} e \frac{\partial \delta u}{\partial n} \, \text{d}s - \int_{\Omega^+} \nabla \cdot (e \nabla \delta u) \, \text{d}x + \int_{\partial \Omega^-} e \frac{\partial \delta u}{\partial n} \, \text{d}s - \int_{\Omega^-} \nabla \cdot (e \nabla \delta u) \, \text{d}x \\
+ \int_{\Omega^+} \kappa^2 \cosh(u^n + u^*) \delta u \, \text{d}x + \int_{\Omega^-} \kappa^2 \cosh(u^n + u^*) \delta u \, \text{d}x
\]

\[
= \int_{\Gamma} \left[ e \frac{\partial \delta u}{\partial n} \right] \, \text{d}s + \sum_{i=+, -} \left( - \int_{\Gamma_i} \nabla \cdot (e \nabla \delta u) \, \text{d}x + \int_{\Omega_i} \kappa^2 \cosh(u^n + u^*) \delta u \, \text{d}x \right) \\
= \int_{\Gamma} \left[ e \frac{\partial \delta u}{\partial n} \right] \, \text{d}s + \int_{\Omega} \nabla \cdot e \nabla (u^n + u^*) \, \text{d}x - \int_{\Omega} \kappa^2 \sinh(u^n + u^*) \, \text{d}x \\
+ \int_{\Omega^-} \kappa^2 \sinh(u^n + u^*) \, \text{d}x
\]

By integrating by parts and using \([u^n] = 0,\)

\[
= \int_{\Gamma} \left[ e \frac{\partial \delta u}{\partial n} \right] \, \text{d}s + \int_{\Omega} \nabla \cdot e \nabla (u^n + u^*) \, \text{d}x - \int_{\Omega} \kappa^2 \sinh(u^n + u^*) \, \text{d}x
\]

Comparing this with the right side of (3.7), we conclude that

\[
\left[ e \frac{\partial \delta u}{\partial n} \right] = 0.
\]

This gives (3.5c). The assumption of (3.7) gives (3.5b) and (3.5d).

Eq. (3.7) has a form of linearized PBE for \(\delta u\) with homogeneous jumps. In the next section, we introduce DB-IFEM to solve the equation.

4 DB-IFEM with consistency terms

From this section on, we assume that \(\Omega\) is a rectangular domain for the simplicity of presentation. Let \(T^*_h\) be a uniform triangulations of \(\Omega\) which do not necessarily have to align with the interface. For example, we divide \(\Omega\) by axi-parallel lines and then subdivide the subrectangles by diagonals. The set of all triangle elements divided by the interface is denoted by \(T^*_h\). We may assume that the interface intersects with the boundary of an element at most twice (see Fig. 3), which is possible when the mesh size is sufficiently small. We define \(S^h(I)\) to be the union of elements in \(T^*_h\). As noted earlier, we can choose \(u^*\) so that the support of \(u^*\) is \(S^h(I)\).

4.1 Modified IFEM for interface problem

First, we review the early version of IFEMs based on \(P_1\)-conforming FEM introduced in [6, 23, 24, 27, 28]. Let \(T\) be any elements in \(T^*_h\). \(S^h(I)\) denotes the space of standard
linear functions on $T$ which has dofs on each node of $T$. Suppose $T$ is cut through edges $e_1$ and $e_2$ at points $E_1$ and $E_2$ by $\Gamma$. Let $T^+$ and $T^-$ be two regions of $T$ separated by $E_1E_2$.

We modify $\phi \in S_h(T)$ so that the new function $\hat{\phi}$ is a piecewise linear on $T$ having the same values at nodes and satisfy jump conditions. Let $L_j (j = 1, 2, 3)$ be linear Lagrange nodal basis functions associated with the vertices $A_j (j = 1, 2, 3)$, i.e., $L_j(A_i) = \delta_{ij}$. Then the form of $\hat{\phi}$ with nodal basis value $V_i = \phi(A_i) (i = 1, 2, 3)$ is following:

\[
\hat{\phi}(x, y) = \begin{cases} 
\hat{\phi}^+ = c_1L_1 + V_2L_2 + V_3L_3, & \text{if } (x, y) \in T^+, \\
\hat{\phi}^- = V_1L_1 + c_2L_2 + c_3L_3, & \text{if } (x, y) \in T^-,
\end{cases}
\]

whose coefficients are determined by the following two homogenous jump conditions, and one homogeneous flux jump condition:

\[
\hat{\phi}^+(E_i) = \hat{\phi}^-(E_i), \quad i = 1, 2, \\
\epsilon^+ \nabla \hat{\phi}^+ \cdot n_{E_1E_2} = \epsilon^- \nabla \hat{\phi}^- \cdot n_{E_1E_2}. 
\]

It is well known that the coefficients of $\hat{\phi}$ is determined uniquely by (4.2a)-(4.2b) [27]. The space of these functions $\hat{\phi}$ is denoted by $\hat{S}_h(T)$. Now we define $P_1$-conforming based IFEM space $\hat{S}_h(\Omega)$ as follows:

\[
\hat{S}_h(\Omega) := \begin{cases} 
\hat{\phi}|_T \in S_h(T), & \text{if } T \text{ is a noninterface element}, \\
\hat{\phi}|_T \in \hat{S}_h(T), & \text{if } T \text{ is an interface element}, \\
\hat{\phi}|_{T_1}(X) = \hat{\phi}|_{T_2}(X), & \text{where } X \text{ is a common node of } T_1 \text{ and } T_2, \\
\hat{\phi}|_T(X) = 0, & \text{where } X \text{ is a node on } \partial \Omega.
\end{cases}
\]

We consider a bilinear form $\tilde{a}_h(\cdot, \cdot)$ on $H_h(\Omega) := H_0^1(\Omega) + \hat{S}_h(\Omega)$ as done in the early version of IFEM [6, 24, 27] as follows:

\[
a_h(\delta u, v) := \sum_{T \in T_h} \left( \int_T \epsilon \nabla \delta u \cdot \nabla v dx + \int_T \kappa^2 \cosh(u^n + u^*) \delta uv dx \right). 
\]
However, as shown by Lin et al. [29], there are cases when the order of accuracy deteriorates. The reason lies in the inconsistency of the bilinear form. Hence we define following consistency terms:

\[ b(\delta u, v) := \sum_{e \in \mathcal{E}_h} \left( - \int_e \{ \epsilon \nabla \delta u \cdot n_e \} e [v]_e \, ds + \int_e \{ \epsilon \nabla v \cdot n_e \} e [\delta u]_e \, ds \right), \]

\[ j(\delta u, v) := \sum_{e \in \mathcal{E}_h} \int_{\partial_h} [\delta u]_e [v]_e \, ds. \]

Then we define new bilinear form adding these consistency terms:

\[ \tilde{a}_h(\delta u, v) := a_h(\delta u, v) + b(\delta u, v) + j(\delta u, v), \quad (4.4) \]

where \( \mathcal{E}_h \) is a set of all edges of \( \mathcal{T}_h \) and \( n_e \) is a fixed directional unit normal vector to each edge. Here, \( \{ v \}_e \) and \( [v]_e \) are the average and jump of \( v \in H_h(\Omega) \) on an edge \( e \in \mathcal{E}_h \) respectively. This technique resembles that of DG.

In a similar manner, we add the consistency terms arising from \( u^n + u^s \) in (3.7). Hence we rewrite (3.7) as follows: Find \( \delta u \in H_h(\Omega) \) satisfying

\[ \tilde{a}(\delta u, v) = \left< J_2, v >_{\Gamma} - e(\nabla (u^n + u^s), \nabla v) - \kappa^2 (\sinh(u^n + u^s), v) \right. \]

\[ \left. - b(u^n + u^s, v) - j(u^n + u^s, v), \quad \forall v \in H_h(\Omega). \right. \quad (4.5) \]

### 4.2 DB-IFEM based on \( P_1 \)-conforming FEM

We briefly explain how to construct DB-IFEM based on \( P_1 \)-conforming FEM introduced in [2]. First, we let \( u^n_h \) be an approximation to \( u^s \). We want that \( u^n_h \) to vanish at each vertex of \( T \) so that \( u^n_h \) do not affect the value of \( \tilde{u} \) at vertices. Hence, \( u^n_h \) is piecewise linear on \( T^+ \) and \( T^- \) and is defined by:

\[ u^n_h(A_i) = 0, \quad i = 1, 2, 3, \quad (4.6a) \]

\[ [u^n_h(E_i)]_{E_1E_2} = J_1(E_i), \quad i = 1, 2, \quad (4.6b) \]

\[ e^+ \frac{\partial \tilde{\phi}^+}{\partial n_{E_1E_2}} - e^- \frac{\partial \tilde{\phi}^-}{\partial n_{E_1E_2}} = \frac{1}{|E_1E_2|} \int_{E_1E_2} J_2. \quad (4.6c) \]

Having constructed \( u^n_h \) (see Fig. 4), we discretize the variational form (4.5). We shall use the new bilinear form (4.4). Then, we define DB-IFEM with consistency terms: Find \( \delta u_h \in \tilde{S}_h(\Omega) \) such that

\[ \tilde{a}_h(\delta u_h, v_h) = \left< J_2, v >_{\Gamma} - e(\nabla (u^n_h + u^s_h), \nabla v_h) - \kappa^2 (\sinh(u^n_h + u^s_h), v_h) \right. \]

\[ \left. - b(u^n_h + u^s_h, v_h) - j(u^n_h + u^s_h, v_h), \quad \forall v_h \in \tilde{S}_h(\Omega). \right. \quad (4.7) \]
Finally, the entire algorithm to obtain the numerical solution $u_h$ of PBE is following:

1) Construct $u_h^*$ by (4.6)
2) Set $u_h^0$, an initial guess
   For $n = 0, 1, 2, \cdots$
      Solve (4.7) for $\delta u_h$
      Set $u_h^{n+1} = u_h^n + \delta u_h$
      If $\|u_h^{n+1} - u_h^n\| < \text{tol}$ stop
   End
3) $\bar{u}_h = u_h^{n+1}$
4) $u_h = \bar{u}_h + u_h^s + u^s$

**Remark 4.1.** For the fitted finite element method for PBE, Xu et al. [4] provided a rigorous convergence analysis using the monotonicity of the operator (after regularizing the problem). However, in our case, the nonhomogeneous jump appears during the regularization process. After removing the jumps, we are left with a regular nonlinear problem with no jump conditions, for which we can combine a standard analysis for nonlinear problem [43] and the analysis of [28] for linear case. Detailed proof will be shown elsewhere.

5 Numerical experiment

In this section, we provide some experiments. The experiments are conducted for problems with various interface $\Gamma$. The typical numerical values of the dielectric coefficient $\epsilon$
in [15] and the Debye-Hückel parameter $\kappa^2$ in [9] are the following:

$$\epsilon^+ \approx 2, \epsilon^- \approx 80,$$

$$\kappa^2 \approx 8.847.$$ We provide the numerical results in terms of $L^2$-error and piecewise $H^1$-error where the piecewise $H^1$-norm is defined as:

$$||\phi||_{1,h}^2 := \sum_{T \in T_h} ||\phi||_{H^1(T)}^2.$$ We let the domain $\Omega = [-1,1]^2$ and we consider a uniform triangulation $T_h$ constructed by right triangles whose size $h = 2^{-k}$, $k = 3, 4, \cdots, 8$. We introduce a level set function $L(x,y) : \Omega \rightarrow \mathbb{R}$ and take the zero level set $\Gamma = \{(x,y) \mid L(x,y) = 0\}$ as the interface which separate the domain $\Omega$ into the subdomain $\Omega^+$ and $\Omega^-$. $L(x,y) = \begin{cases} < 0, & (x,y) \in \Omega^-, \\ 0, & (x,y) \in \Gamma, \\ > 0, & (x,y) \in \Omega^+. \end{cases}$

In each example, we use various $\epsilon$ and $\kappa^2$ to test the applicability of our scheme. The tolerance of Newton iteration is set by $10^{-9}$. Example 5.1. (Star shape) The interface is given by $\Gamma = \{(r, \theta) : r - (0.57 + 0.1(\sin(4\theta))) = 0\}$. Here we take $\kappa^2 = 0.1$, $\epsilon^- = 50$ and $\epsilon^+ = 1$. The singular charges are located on $x_i = \pm 0.2$. The exact regular solution $u^r$ is

$$u^r = \begin{cases} x^2 + y^2, & \text{on } \Omega^- \\ 0.64 \exp(-x^2 - y^2), & \text{on } \Omega^+. \end{cases} \tag{5.1}$$ We report the $L^2$ and $H^1$-errors of DB-IFEM in Table 1. We observe that the convergence rates are both optimal in $L^2$ and $H^1$-norms. The graph is given in Fig. 5.

| $1/h$ | $||u - u_h||_{L^2(\Omega)}$ order | $||u - u_h||_{1,h}$ order | iterations |
|------|---------------------------------|--------------------------|------------|
| 8    | $5.431 \times 10^{-3}$          | $1.516 \times 10^{-1}$    | 3          |
| 16   | $1.139 \times 10^{-3}$          | $6.651 \times 10^{-2}$    | 3          |
| 32   | $2.761 \times 10^{-4}$          | $3.183 \times 10^{-2}$    | 1.063      | 3          |
| 64   | $6.174 \times 10^{-5}$          | $1.521 \times 10^{-2}$    | 1.066      | 3          |
| 128  | $1.602 \times 10^{-5}$          | $7.463 \times 10^{-3}$    | 1.027      | 3          |
| 256  | $3.897 \times 10^{-6}$          | $3.692 \times 10^{-3}$    | 1.016      | 3          |
Example 5.2. (Ellipse) The interface is given by \( \Gamma = \{(x,y) : \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0\} \) where \( a = 0.61 \) and \( b = 0.41 \). Here we take \( \kappa^2 = 10, \epsilon^- = 1 \) and \( \epsilon^+ = 1000 \). The singular charges are located on \( x_i = 0 \). The exact regular solution \( u^r \) is
\[
  u^r = \begin{cases} 
    x^2 - y^2 + 3, & \text{on } \Omega^-, \\
    1 - x^2 - y^2, & \text{on } \Omega^+. 
  \end{cases}
\] (5.2)
We report the \( L^2 \) and \( H^1 \)-errors of DB-IFEM in Table 2. We observe that the convergence rates are both optimal in \( L^2 \) and \( H^1 \)-norms. (For the graph, see Fig. 6.)

<table>
<thead>
<tr>
<th>( 1/h )</th>
<th>( | u - u_h |_{L^2(\Omega)} ) order</th>
<th>( | u - u_h |_{1,h} ) order</th>
<th>iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>( 8.373 \times 10^{-2} )</td>
<td>( 2.418 \times 10^{-1} )</td>
<td>3</td>
</tr>
<tr>
<td>16</td>
<td>( 4.069 \times 10^{-3} )</td>
<td>( 4.363 \times 10^{-1} )</td>
<td>0.94</td>
</tr>
<tr>
<td>32</td>
<td>( 9.031 \times 10^{-4} )</td>
<td>( 2.172 \times 10^{-1} )</td>
<td>1.08</td>
</tr>
<tr>
<td>64</td>
<td>( 1.822 \times 10^{-4} )</td>
<td>( 2.309 \times 10^{-2} )</td>
<td>1.10</td>
</tr>
<tr>
<td>128</td>
<td>( 4.369 \times 10^{-5} )</td>
<td>( 2.061 \times 10^{-2} )</td>
<td>1.04</td>
</tr>
<tr>
<td>256</td>
<td>( 1.022 \times 10^{-5} )</td>
<td>( 2.095 \times 10^{-3} )</td>
<td>1.036</td>
</tr>
</tbody>
</table>

Example 5.3. (Circle) The interface is given by \( \Gamma = \{(x,y) : x^2 + y^2 - 0.6111^2 = 0\} \). Here we take \( \kappa^2 = 50, \epsilon^- = 1 \) and \( \epsilon^+ = 100 \). The singular charges are located on \( (x_i,y_i) = (0,\pm 0.5), (\pm 0.5,0) \). The exact regular solution \( u^r \) is
\[
  u^r = \begin{cases} 
    \sin(xy), & \text{on } \Omega^-, \\
    \cos\left(\frac{x^2+y^2}{2}\right), & \text{on } \Omega^+. 
  \end{cases}
\] (5.3)
Figure 6: The shape of the domain and the graph of numerical solution of $h = 6$.

Figure 7: The shape of the domain and the graph of numerical solution of $h = 6$.

We report the $L^2$ and $H^1$-errors of DB-IFEM in Table 3. We observe that the convergence rates are both optimal in $L^2$ and $H^1$-norms. (For the graph, see Fig. 7.)

<table>
<thead>
<tr>
<th>$1/h$</th>
<th>$| \mathbf{u} - \mathbf{u}<em>h |</em>{L^2(\Omega)}$ order</th>
<th>$| \mathbf{u} - \mathbf{u}<em>h |</em>{H^1}$ order</th>
<th>iterations</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>$3.552 \times 10^{-2}$ 4.993</td>
<td>$1.210 \times 10^{-1}$ 1.047</td>
<td>4</td>
</tr>
<tr>
<td>16</td>
<td>$1.163 \times 10^{-3}$ 2.047</td>
<td>$5.860 \times 10^{-2}$ 1.007</td>
<td>4</td>
</tr>
<tr>
<td>32</td>
<td>$2.814 \times 10^{-4}$ 2.037</td>
<td>$2.915 \times 10^{-2}$ 1.022</td>
<td>4</td>
</tr>
<tr>
<td>64</td>
<td>$6.153 \times 10^{-5}$ 2.037</td>
<td>$1.436 \times 10^{-2}$ 1.022</td>
<td>4</td>
</tr>
<tr>
<td>128</td>
<td>$1.499 \times 10^{-5}$ 2.037</td>
<td>$7.140 \times 10^{-3}$ 1.008</td>
<td>4</td>
</tr>
<tr>
<td>256</td>
<td>$3.750 \times 10^{-6}$ 1.999</td>
<td>$3.563 \times 10^{-3}$ 1.003</td>
<td>4</td>
</tr>
</tbody>
</table>
Example 5.4. In this example, we consider an actual simulation of PBE when the singular charges are very close to the interface. The interface is given by $\Gamma = \{(x, y): x^2 + y^2 - 0.56^2 = 0\}$. Here we take $\kappa^2 = 8.847$, $\epsilon^- = 2$ and $\epsilon^+ = 80$. The singular charges are located on $(x_i, y_i) = (\pm 0.5, 0)$. In this case, the exact solution is not known. We observe that the numerical solution seems to converge well without any instability. (For the graph, see Fig. 8.) We see similar behavior when the singular charges are closer to the interface, as long as the mesh resolves the singularity.

Figure 8: The location of singular charge and the graph of numerical solution of $\varepsilon = 6$. 
6 Conclusion

In this work, we introduced a new numerical method to solve the PBE. First, we regularized the solution of the PBE. We introduced DB to treat nonhomogeneous jumps and applied linearization to the equation before discretization. Next, we defined a new variational form. Then, we discretize the DB and the bilinear form of the PBE with added consistency terms. Finally, we solved the discretized problem by IFEM. We repeated this process. We believe that this kind of scheme for removing the singularity of the PBE and handling nonhomogeneous jumps of regular component is unique. In the numerical experiments, we observed optimal convergence rates for all the examples.

Acknowledgments

This work is supported by NRF (Contract no. 2017R1D1A1B03032765).

References

pp. 2–4.


