A First-Order System Least-Squares Finite Element Method for the Poisson-Boltzmann Equation

STEPHEN D. BOND,[1] JEHANZEB HAMEED CHAUDHRY,[1] ERIC C. CYR,[2] LUKE N. OLSON[1]
[1]*Department of Computer Science, University of Illinois, Urbana, Illinois 61801*
[2] *Department of Scalable Algorithms, Sandia National Laboratory, Albuquerque, New Mexico, 87185*

Received 1 June 2009; Revised 18 August 2009; Accepted 23 September 2009
DOI 10.1002/jcc.21446
Published online in Wiley InterScience (www.interscience.wiley.com).

Abstract: The Poisson-Boltzmann equation is an important tool in modeling solvent in biomolecular systems. In this article, we focus on numerical approximations to the electrostatic potential expressed in the regularized linear Poisson-Boltzmann equation. We expose the flux directly through a first-order system form of the equation. Using this formulation, we propose a system that yields a tractable least-squares finite element formulation and establish theory to support this approach. The least-squares finite element approximation naturally provides an *a posteriori* error estimator and we present numerical evidence in support of the method. The computational results highlight optimality in the case of adaptive mesh refinement for a variety of molecular configurations. In particular, we show promising performance for the Born ion, Fasciculin 1, methanol, and a dipole, which highlights robustness of our approach.

 J Comput Chem 00: 000–000, 2010

Key words: Poisson-Boltzmann; implicit solvent; finite elements; least-squares; adaptive refinement

Introduction

Solvent plays a critical role in determining the structure and function of biomolecular systems. However, the explicit representation of solvent at a molecular level is often intractable because of the range of scales required. Moreover, properly modeling solvent interactions with molecules are computationally expensive because of the complexity of the atomistic interactions that must be sampled over multiple configurations. As such, implicit solvent models, such as the Poisson-Boltzmann model[1] and Generalized Born model,[2] confront this difficulty by treating the solvent as a bulk continuum.

The focus of this work is on numerical solutions to the Poisson-Boltzmann equation (PBE), which approximates the mean solvent forces by assuming the ions are distributed according to the Boltzmann distribution. This results in a unique electrostatic potential described by this implicit solvent model.[3] In particular, we seek a numerical solution of the linearization of the regularized PBE (RPBE). The use of a regularized formulation[3] is required because the original statement of the PBE yields singularities in the electrostatic potential. Regularization overcomes this issue by analytically subtracting the singularities from the electrostatic potential yielding a modified version of the original PDE. To further simplify the problem and focus on the efficacy of our discretization, we linearize the RPBE. The linearized version has many of the same challenges as the RPBE; however, it features reduced computation cost[4] when remaining as a physically accurate perturbation to the fully nonlinear problem.[5]

A number of different directions for numerically solving the Poisson-Boltzmann equation have been pursued. Approaches such as finite difference and finite volume methods,[6–15] finite elements methods,[3,16–22] boundary element methods,[23–34] and integral equations[35,36] have been developed for this problem. Yet, as the complexity of applications increases so do the demands on the numerical approximation, and we are motivated to investigate additional computational tools that provide a medium for more robust and efficient simulation.

In this article, we focus on a variational setting for the PBE because of the underlying theoretical support for numerical methods and the established analysis of the equation. In particular, we propose a least-squares finite element formulation of the linear regularized Poisson-Boltzmann equation. Least-squares finite element methods offer a viable approach to efficient and accurate approximation. The least-squares method that we follow begins by reforming the partial differential equation as a first-order system. A functional is then constructed based on the residual equations of the first-order system and is minimized. A first-order system least-squares

Correspondence to: J. H. Chaudhry; e-mail: jhameed2@illinois.edu

Contract/grant sponsor: NSF-CCF; contract/grant number: 08-30578

Contract/grant sponsor: NSF-DMS; contract/grant number: 07-46676

Contract/grant sponsor: University of Illinois CSE Fellowship

Contract/grant sponsor: Sandia is sponsored under U.S. Department of Energy; contract/grant number: DE-AC04-94AL85000

(FOSLS) approach to finite elements has shown to be effective for numerous problems. In particular, elliptic problems[37,38] with discontinuous coefficients[39–41] are theoretically competitive and numerically plausible.

The existing FOSLS theory motivates our treatment of the PBE; yet, the theoretical properties for the FOSLS form we pose in "FOSLS formulation of PBE" Section are not fully developed. We establish these results and confirm the existence of a unique solution for our problem. We propose a first-order system for the PBE in "FOSLS formulation of PBE" Section that correctly addresses the jump discontinuity inherent in the problem. The PBE is described through a dielectric coefficient, $\epsilon(x)$, and Debye-Hückel parameter, $\bar{\kappa}(x)$, that are discontinuous across an interface. Proper treatment of the flux term across this interface is critical to the variational formulation. To this end, we propose a unique form of the flux that both captures the underlying physics and yields a system amenable to a least-squares minimization.

The goal of this article is to outline a least-squares finite element method for use with existing computational tools, such as the Finite Element Toolkit (FETk),[42] which uses piecewise linear elements over tetrahedral tessellations of single domains. The result is a competitive and straightforward finite element method for the PBE using adaptive mesh refinement. Adaptive refinement using finite elements has been studied for the Poisson-Boltzmann equation in a Galerkin formulation.[18,19] These approaches focus on resolution of the singularities in the original PDE. Here, we use the functional provided by the least-squares formulation to guide refinement with similar success. Treatment of the interface condition is automatic in our formulation of the problem, naturally capturing the physics around the interface when still being amenable to approximation by standard finite elements.

The remainder of the article is organized as follows. In Section on Poisson Boltzmann Equation, we summarize the PBE, its regularization and linearization, and the general problem domain. We outline the FOSLS terminology in Section on FOSLS formulation of PBE and introduce our formulation of the method. Moreover, we establish theoretically the use our formulation and discuss implications and techniques for computational simulation. In Section on Numerical Experiments, we provide numerical evidence of effectiveness of the FOSLS approach for a number of molecular systems. The method is shown to be effective for problems with known solutions (Born ion), for more complicated structures (Fasciculin 1 and methanol), and for a problem with low regularity (dipoles).

Poisson Boltzmann Equation

The Poisson-Boltzmann equation models the electrostatic activity between molecules in an ionic solvent. In this model, it is assumed that the ions in the solvent are distributed according to the Boltzmann distribution and that the potential of the mean force on a particle is simply the charge of the ion times the electrostatic potential. This yields the general Poisson-Boltzmann equation,[5]

$$-\nabla\cdot(\epsilon(x)\nabla\phi(x)) = 4\pi\rho_{\mathrm{f}}(x) + 4\pi\sum_{j=1}^{n_{\mathrm{s}}} c_j^{\mathrm{s}} Q_j^{\mathrm{s}}\lambda_j(x) \times \exp\left[\frac{-1}{k_B T}Q_j^{s}\phi(x)\right], \quad (1a)$$

$$\lim_{\|x\|\to\infty}\phi(x) = 0. \quad (1b)$$

Here, ϕ is the unknown electrostatic potential, ϵ is the dielectric coefficient, ρ_{f} is the fixed charge distribution in the solute (biomolecule), k_B is the Boltzmann constant, and T is the temperature. It is assumed that the solvent is composed of n_{s} species of ions, each with charge Q_j^s and concentration c_j^s. The accessibility of the jth ion-species to a point, x, in space is described by $\lambda_j(x)$.

For a solute in a 1:1 electrolyte solvent (e.g., NaCl), the charge of each ion species is ± 1 unit charge, and the general Poisson-Boltzmann equation simplifies[1] to

$$-\nabla\cdot(\epsilon(x)\nabla\phi(x)) + \bar{\kappa}^2(x)\left(\frac{k_B T}{e_{\mathrm{c}}}\right)\sinh\left(\frac{e_{\mathrm{c}}\phi(x)}{k_B T}\right) = 4\pi\sum_{i=1}^{m} Q_i\delta(x-x_i), \quad (2a)$$

$$\lim_{\|x\|\to\infty}\phi(x) = 0. \quad (2b)$$

Here, we have further assumed that solute contains a total of m fixed point charges, with the ith charge, Q_i, centered at position x_i. The resulting distribution, ρ_{f}, is a linear combination of Dirac delta functions, $\delta(x-x_i)$.

The domain for the problem, $\mathbb{R}^3$, is subdivided into a molecular region, Ω_{m}, a solvent region, $\Omega_{\mathrm{s}}^{\infty}$, and an interface between the two, denoted by Γ. The solute is surrounded by solvent, which is represented as a continuum over the subdomain $\Omega_{\mathrm{s}}^{\infty} = \mathbb{R}^3\backslash\overline{\Omega}_{\mathrm{m}}$. In some Poisson-Boltzmann models, an additional ion exclusion or Stern layer is present between Ω_{s} and Ω_{m}. The Stern layer provides separation between the solute and the ions of the solvent. As a result, the dielectric matches the dielectric in the solvent region and the ionic strength is zero ($\bar{\kappa} = 0$). In this article, we focus on the more challenging issue of the jump in the dielectric, and neglect the Stern layer. The subdomains for a typical biomolecular solute are shown in Figure 1. The dielectric coefficient, $\epsilon(x)$, and modified

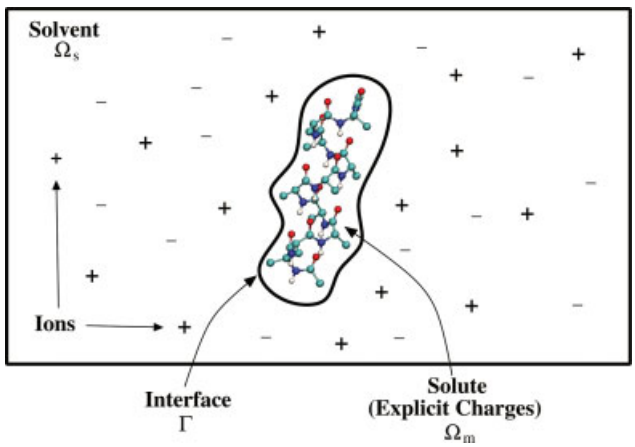


Figure 1. Subdomains for the Poisson-Boltzmann equation. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

Debye-Hückel parameter, $\bar{\kappa}(x)$, describe the accessibility of the solvent to the solute and are defined on $\Omega_m \cup \Omega_s^\infty$ by the piecewise constant functions

$$\epsilon(x) = \begin{cases} \epsilon_m & x \in \Omega_m \\ \epsilon_s & x \in \Omega_s^\infty \end{cases} \quad \text{and}$$

$$\bar{\kappa}^2(x) = \begin{cases} 0 & x \in \Omega_m \\ \bar{\kappa}_s^2 = \epsilon_s \frac{8\pi N_A e_c^2}{1000 k_B T} I_s & x \in \Omega_s^\infty \end{cases}. \tag{3}$$

Here, ϵ_m and ϵ_s are positive constants, N_A is Avogadro's number, and e_c is the charge of a proton. The ionic strength, I_S, is a physical parameter which varies depending on the solvent.

For computational reasons, the unbounded solvent domain, Ω_s^∞, is typically truncated at a finite radius from the "center" of the molecule, which gives rise to a bounded solvent domain, Ω_s. Dirichlet boundary conditions are imposed to capture the asymptotic behavior of the solution on an unbounded domain. Combining this with the change of variables, $\tilde{u}(x) = e_c\phi(x)/k_BT$, results in a dimensionless Poisson-Boltzmann equation on the spherical domain $\Omega = \Omega_m \cup \Omega_s \cup \Gamma$:

$$-\nabla \cdot (\epsilon(x)\nabla\tilde{u}(x)) + \bar{\kappa}^2(x)\sinh\tilde{u}(x) = \frac{4\pi e_c}{k_BT}\sum_{i=1}^{m} Q_i\delta(x - x_i), \quad x \in \Omega_m \cup \Omega_s, \tag{4a}$$

$$\tilde{u}(x) = g(x), \quad x \in \partial\Omega_s, \tag{4b}$$

$$\left[\!\left[\epsilon(x)\frac{\partial\tilde{u}(x)}{\partial\mathbf{n}}\right]\!\right]_\Gamma = 0, \quad x \in \Gamma. \tag{4c}$$

where the jump at the interface is defined as

$$\left[\!\left[\epsilon(x)\frac{\partial\tilde{u}(x)}{\partial\mathbf{n}}\right]\!\right]_\Gamma = \lim_{\alpha\to 0^+} \epsilon(x+\alpha\mathbf{n})\frac{\partial\tilde{u}(x+\alpha\mathbf{n})}{\partial\mathbf{n}} - \epsilon(x-\alpha\mathbf{n})\frac{\partial\tilde{u}(x-\alpha\mathbf{n})}{\partial\mathbf{n}},$$

with $\mathbf{n}$ as the unit normal to the interface Γ.

The boundary conditions are prescribed using a linear combination of Helmholtz Green's functions,

$$g = \frac{e_c}{k_BT}\sum_{i=1}^{m}\frac{Q_i}{\epsilon_s|x-x_i|}\exp\left(\frac{-\bar{\kappa}_s|x-x_i|}{\sqrt{\epsilon_s}}\right). \tag{5}$$

In contrast to (2a), the principal equation is defined over each subdomain and an interface condition is introduced on Γ. This restatement makes explicit the normal continuity implied by the strong form divergence of $\epsilon(x)\nabla\phi$ across the interface Γ in the original PBE.

We denote the standard Sobolev spaces as $L^2(\Omega)$ and $H^k(\Omega)$, for $k \geq 0$. $H^k(\Omega)$ consists of functions over Ω having square integrable (weak) derivatives of order up to k. The norms on $L^2(\Omega)$ and $H^k(\Omega)$ are expressed as $\|\cdot\|_{0,\Omega}$ and $\|\cdot\|_{k,\Omega}$, with the $L^2(\Omega)$ inner product written $(\cdot,\cdot)_{0,\Omega}$. In addition, we define the Hilbert spaces

$$H(\mathrm{div};\Omega) := \{\mathbf{q} \in L^2(\Omega)^3 : \nabla\cdot\mathbf{q} \in L^2(\Omega)\},$$
$$H_0(\mathrm{div};\Omega) := \{\mathbf{q} \in H(\mathrm{div};\Omega) : \mathbf{n}\times\mathbf{q} = 0 \text{ on } \partial\Omega\},$$
$$H_0^1(\Omega) := \{u \in H^1(\Omega) : u = 0 \text{ on } \partial\Omega\},$$

with norms

$$\|\mathbf{q}\|_{\mathrm{div},\Omega}^2 = \|\mathbf{q}\|_{0,\Omega}^2 + \|\nabla\cdot\mathbf{q}\|_{0,\Omega}^2, \tag{6}$$
$$\|u\|_{1,\Omega}^2 = \|u\|_{0,\Omega}^2 + \|\nabla u\|_{0,\Omega}^2. \tag{7}$$

One difficulty with (2a) is regularity. The right-hand side $4\pi\sum_i Q_i\delta(x-x_i)$ is not in $H^{-1}(\Omega)$, i.e., the dual space of $H_0^1(\Omega)$. Practically, the right-hand side induces singularities in $\tilde{u}$ at the solute atom centers x_i. These singularities are the familiar consequence of solute–solute electrostatic interactions satisfying Coulomb's law. However, finite element and finite difference methods often require more smoothness to guarantee convergence. Following ref. 3, we overcome this issue by decomposing $\tilde{u}$ into

$$\tilde{u} = u + u_c, \tag{8}$$

where u is an unknown smooth function and u_c is a known singular function. The Coulomb function, u_c, satisfies the Poisson equation

$$-\epsilon_m\nabla\cdot\nabla u_c(x) = \frac{4\pi e_c}{k_BT}\sum_{i=1}^{m} Q_i\delta(x-x_i), \tag{9}$$

and absorbs the singularities in $\tilde{u}$. Combining (8) with (4), we obtain the regularized PBE or RPBE

$$\begin{aligned} -\nabla\cdot\epsilon(x)\nabla u(x) + \bar{\kappa}^2(x)\sinh(u(x) + u_c(x)) &= \nabla\cdot(\epsilon(x)-\epsilon_m)\nabla u_c(x), && x \in \Omega_s \cup \Omega_m, \\ u(x) &= g(x) - u_c(x), && x \in \partial\Omega, \\ \left[\!\left[\epsilon(x)\frac{\partial u(x)}{\partial\mathbf{n}}\right]\!\right]_\Gamma &= (\epsilon_m - \epsilon_s)\frac{\partial u_c(x)}{\partial\mathbf{n}}, && x \in \Gamma. \end{aligned} \tag{10}$$

As $\bar{\kappa}(x)$ and $\epsilon(x) - \epsilon_m$ are zero inside Ω_m, we avoid evaluating the Coulomb potential, u_c, near the singularities present at each point charge, $x_i \in \Omega_m$. This yields a right-hand side in (10) that is a well-defined distribution in $H^{-1}(\Omega)$ and, as a result, eq. (10) is a well-defined nonlinear second-order elliptic equation with a unique weak solution u in $H^1(\Omega)$.[3]

A simplified version of (10) is the linear regularized Poisson-Boltzmann equation, which is obtained by linearizing the hyperbolic sine:

$$-\nabla\cdot\epsilon(x)\nabla u(x)+\bar{\kappa}^2(x)u(x) = \nabla\cdot(\epsilon(x)-\epsilon_{\mathrm{m}})\nabla u_{\mathrm{c}}(x)-\bar{\kappa}^2(x)u_{\mathrm{c}}(x),\quad x\in\Omega_{\mathrm{s}}\cup\Omega_{\mathrm{m}}, \tag{11a}$$

$$u(x)=g(x)-u_{\mathrm{c}}(x),\quad x\in\partial\Omega, \tag{11b}$$

$$\left[\!\left[\epsilon(x)\frac{\partial u(x)}{\partial\mathbf{n}}\right]\!\right]_{\Gamma}=(\epsilon_{\mathrm{m}}-\epsilon_{\mathrm{s}})\frac{\partial u_{\mathrm{c}}(x)}{\partial\mathbf{n}},\quad x\in\Gamma. \tag{11c}$$

Physically, the linearization reduces the ionic response of the solvent to the solute. This approximation is acceptable unless the solute is highly charged.[4] In this case, the magnitude of the electrostatic potential is large and the approximation $\sinh(u)\approx u$ is not accurate.[1]

FOSLS Formulation of PBE

The First-Order System Least Squares (FOSLS) finite element method is an alternative to standard and mixed Galerkin finite element methods.[43] FOSLS begins by converting the PDE to a first order system. Using the new set of equations, a functional is then defined whose minimizer solves the original PDE.

FOSLS offers a number of potential advantages over traditional methods. The functional is minimized using a variational principle, giving rise to a symmetric bilinear form. A discretization based on this form leads to a symmetric positive-definite linear system, which is ideal for solvers such as preconditioned conjugate gradient. Also, the bilinear form is often elliptic with respect to a practical norm, and as a result the finite element spaces do not need to satisfy the discrete inf-sup condition of Ladyzhenskaya-Babuŝka-Brezzi,[44] unlike mixed methods. A practical consequence is that basic finite element spaces, e.g., continuous piecewise linear polynomials, may be used for all variables.

The FOSLS functional also provides a local *a posteriori* error estimate. Such estimates are complicated for other methods, but the FOSLS residual norm provides a straightforward and accurate estimate for our problem. This local error estimate is used for adaptively refining a mesh in our numerical experiments and we highlight the effectiveness of this tool.

Least-squares finite element methods are not without limitation; however, the introduction of new variables to formulate the first-order system ultimately increases the degrees of freedom and complexity in computing the solution. This is not necessarily a disadvantage as the new variables are often physically meaningful and are often needed elsewhere in the simulation. For example, the FOSLS formulation of the PBE introduces a secondary "flux" variable, which is used to effectively compute potential of the mean force required in the solution of the Steady-State Smoluchowski Equation.[45] Another potential drawback is that FOSLS requires more regularity than might be present in the problem to ensure optimal error estimates. Optimal error estimates using a Least-squares approach for PBE can be derived through a multidomain approach for such problems.[46] In this article, we also use adaptive refinement to overcome these issues of computational complexity, yielding optimal convergence rates in our numerical experiments.

A typical approach to forming a first-order system of (11) over a single domain Ω is to introduce a flux, $\tilde{\mathbf{q}}=\epsilon(x)\nabla u$ (e.g., see ref. 39). The resulting first-order system is

$$\begin{aligned}\tilde{\mathbf{q}}-\epsilon\nabla u&=0 && \text{in }\Omega, && (12a)\\ -\nabla\cdot\tilde{\mathbf{q}}+\bar{\kappa}^2u&=\nabla\cdot(\epsilon(x)-\epsilon_{\mathrm{m}})\nabla u_{\mathrm{c}}-\bar{\kappa}^2u_{\mathrm{c}} && \text{in }\Omega, && (12b)\\ u&=g-u_{\mathrm{c}} && \text{on }\partial\Omega. && (12c)\end{aligned}$$

An application of Green's theorem on this system shows that across any surface in Ω with normal $\mathbf{n}$, $\mathbf{n}\cdot\tilde{\mathbf{q}}$ is continuous. In particular, solution to system (12) satisfies,

$$\left[\!\left[\tilde{\mathbf{q}}\cdot\mathbf{n}\right]\!\right]_{\Gamma}=0\quad x\in\Gamma.$$

However, as $\tilde{\mathbf{q}}=\epsilon\nabla u$, eq. (11c) implies,

$$\left[\!\left[\tilde{\mathbf{q}}\cdot\mathbf{n}\right]\!\right]_{\Gamma}=(\epsilon_{\mathrm{m}}-\epsilon_{\mathrm{s}})\nabla u_{\mathrm{c}}(x)\cdot\mathbf{n}\quad x\in\Gamma.$$

This implies $\tilde{\mathbf{q}}\cdot\mathbf{n}$ is not continuous across the interface Γ, and hence, a least squares approach based on system (12) is an incorrect formulation for solving the RPBE.

For a well-posed FOSLS formulation to system (11), we need to define a first-order variable $\mathbf{q}$, whose normal component is not only continuous across the interface but also satisfies the interface condition required by RPBE. To ensure these conditions, we define $\mathbf{q}=\epsilon(x)\nabla u+(\epsilon(x)-\epsilon_{\mathrm{m}})\nabla u_{\mathrm{c}}$, which results in,

$$\begin{aligned}\mathbf{q}/\epsilon(x)-\nabla u&=((\epsilon(x)-\epsilon_{\mathrm{m}})/\epsilon(x))\nabla u_{\mathrm{c}} && \text{in }\Omega, && (13a)\\ -\nabla\cdot\mathbf{q}+\bar{\kappa}^2u&=-\bar{\kappa}^2u_{\mathrm{c}} && \text{in }\Omega, && (13b)\\ u&=g-u_{\mathrm{c}} && \text{on }\partial\Omega, && (13c)\\ \mathbf{n}\times\mathbf{q}&=\mathbf{n}\times(\epsilon_{\mathrm{s}}\nabla g+(\epsilon(x)-\epsilon_{\mathrm{m}})\nabla u_{\mathrm{c}}) && \text{on }\partial\Omega. && (13d)\end{aligned}$$

Now eqs. (11c) and (13) imply,

$$\left[\!\left[\mathbf{q}\cdot\mathbf{n}\right]\!\right]_{\Gamma}=0\quad x\in\Gamma.$$

We now pose our problem in abstract form and establish a unique solution. To simplify the analysis, we consider homogeneous Dirichlet boundary conditions. Using a standard lifting argument, we obtain

$$\begin{aligned}\mathbf{q}/\epsilon(x)-\nabla u&=((\epsilon(x)-\epsilon_{\mathrm{m}})/\epsilon(x))\nabla u_{\mathrm{c}} && \text{in }\Omega,\\ -\nabla\cdot\mathbf{q}+\bar{\kappa}^2u&=-\bar{\kappa}^2u_{\mathrm{c}}, && \text{in }\Omega, && (14)\\ u&=0 && \text{on }\partial\Omega,\\ \mathbf{n}\times\mathbf{q}&=0 && \text{on }\partial\Omega.\end{aligned}$$

The least-squares functional based on (13) is as follows. For $\mathbf{q} \in H_0(\mathrm{div};\Omega)$ and $u \in H_0^1(\Omega)$, we define

$$G(\mathbf{q},u;u_c) = \|\mathbf{q}/\epsilon(x) - \nabla u - ((\epsilon(x)-\epsilon_m)/\epsilon(x))\nabla u_c\|_{0,\Omega}^2 + \| - \nabla\cdot\mathbf{q} + \bar{\kappa}^2 u + \bar{\kappa}^2 u_c\|_{0,\Omega}^2. \quad (15)$$

The solution of (13) solves the minimization problem

$$G(\mathbf{q},u;u_c) = \min_{(\mathbf{r},v)\in H_0(\mathrm{div};\Omega)\times H_0^1(\Omega)} G(\mathbf{r},v;u_c) \quad (16)$$

and leads to the variational problem

$$\mathcal{F}(\mathbf{q},u;\mathbf{r},v) = \ell(\mathbf{r},v), \quad (17)$$

where the bilinear form $\mathcal{F}$ and linear functional ℓ are

$$\mathcal{F}(\mathbf{q},u;\mathbf{r},v) = (\mathbf{q}/\epsilon - \nabla u, \mathbf{r}/\epsilon - \nabla v)_{0,\Omega} + (-\nabla\cdot\mathbf{q} + \bar{\kappa}^2 u, -\nabla\cdot\mathbf{r} + \bar{\kappa}^2 v)_{0,\Omega}, \quad (18)$$

$$\ell(\mathbf{r},v) = -(\bar{\kappa}^2 u_c, -\nabla\cdot\mathbf{r} + \bar{\kappa}^2 v)_{0,\Omega} + (((\epsilon-\epsilon_m)/\epsilon)\nabla u_c, \mathbf{r}/\epsilon - \nabla v)_{0,\Omega}. \quad (19)$$

Ellipticity of FOSLS Functional

To show the variational problem (17) is well-posed, it is sufficient to prove that $G(\mathbf{q},u;0)^{\frac{1}{2}}$ defines a norm equivalent to the $H(\mathrm{div})\times H^1$ norm (Theorem 1). This result also ensures that our finite element solution is the best approximation to the true solution under the norm defined by $G(\mathbf{q},u;0)$. Before proving this norm equivalence, we start by stating and proving a lemma, which will be used in the proof of Theorem 1.

Lemma 1. *Let $h(x)$ and $k(x)$ be two positive bounded functions on Ω, i.e., $0 < c_1 < h(x) < c_2$ and $0 < c_1 < k(x) < c_2$ for all $x \in \Omega$, where c_1 and c_2 are constants. Then there exists positive constants α_1 and α_2 such that*

$$\alpha_1\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u) \le \mathcal{F}(\mathbf{q},u;\mathbf{q},u) \le \alpha_2\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u), \quad (20)$$

where the bilinear form $\hat{\mathcal{F}}$ is defined as

$$\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{r},v) = (\sqrt{h}(\mathbf{q}/\epsilon - \nabla u), \sqrt{h}(\mathbf{r}/\epsilon - \nabla v))_{0,\Omega} + (\sqrt{k}(-\nabla\cdot\mathbf{q} + \bar{\kappa}^2 u), \sqrt{k}(-\nabla\cdot\mathbf{r} + \bar{\kappa}^2 v))_{0,\Omega}. \quad (21)$$

Proof: Taking $\alpha_1 = c_2^{-1}$ and $\alpha_2 = c_1^{-1}$ gives the desired result. ∎

Theorem 1. *The bilinear form $\mathcal{F}$ defines a norm equivalent to the $H(\mathrm{div})\times H^1$ norm. That is, there exists positive constants γ_1 and γ_2 such that*

$$\mathcal{F}(\mathbf{q},u;\mathbf{r},v) \le \gamma_1(\|\mathbf{q}\|_H^2(\mathrm{div}) + \|u\|_{1,\Omega}^2)^{1/2}(\|\mathbf{r}\|_H^2(\mathrm{div}) + \|v\|_{1,\Omega}^2)^{1/2} \quad (22)$$

and

$$\mathcal{F}(\mathbf{q},u;\mathbf{q},u) \ge \gamma_2(\|\mathbf{q}\|_H^2(\mathrm{div}) + \|u\|_{1,\Omega}^2). \quad (23)$$

Proof: A proof for the general case is given in [37]. Here we offer a proof for our specific case, to obtain sharper constants of ellipticity; our proof is in the same spirit as a proof presented in [39].

First we prove boundedness of $\mathcal{F}$ [eq. (22)]. An application of Cauchy-Bunyakovsky-Schwarz inequality to (18) leads to

$$\mathcal{F}(\mathbf{q},u;\mathbf{r},v) \le (\mathcal{F}(\mathbf{q},u;\mathbf{q},u))^{1/2}(\mathcal{F}(\mathbf{r},v;\mathbf{r},v))^{1/2}. \quad (24)$$

Using the fact that ϵ is bounded away from zero in Ω yields

$$\begin{aligned}\mathcal{F}(\mathbf{q},u;\mathbf{q},u) &= \|\mathbf{q}/\epsilon - \nabla u\|_{0,\Omega}^2 + \| - \nabla\cdot\mathbf{q} + \bar{\kappa}^2 u\|_{0,\Omega}^2 \\ &\le \gamma_3(\|\mathbf{q}\|_{0,\Omega}^2 + \|\nabla u\|_{0,\Omega}^2 + \|\nabla\cdot\mathbf{q}\|_{0,\Omega}^2 + \|u\|_{0,\Omega}^2) \\ &= \gamma_3(\|\mathbf{q}\|_H^2(\mathrm{div}) + \|u\|_{1,\Omega}^2), \end{aligned} \quad (25)$$

where $\gamma_3 = \max(2, 2\bar{\kappa}^4, 2\epsilon^{-2}) = \max(2, 2\bar{\kappa}_s^4, 2\epsilon_m^{-2}, 2\epsilon_s^{-2})$. Combining eqs. (25) and (24) proves boundedness of $\mathcal{F}$.

To prove coercivity, we consider a modified bilinear form, as defined by (21). We define $h(x)$ and $k(x)$ as:

$$h(x) = \begin{cases}\epsilon(x) & x\in\Omega_m \\ \tau\epsilon(x) & x\in\Omega_s\end{cases} \quad \text{and} \quad k(x) = \begin{cases}1 & x\in\Omega_m \\ \tau/\bar{\kappa}_s^2 & x\in\Omega_s,\end{cases} \quad (26)$$

where τ is a constant such that $0 < \tau < 1$.

We can decompose the integral over Ω and evaluate $\mathcal{F}$ over Ω_s and Ω_m,

$$\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u) = \hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u)|_{\Omega_s} + \hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u)|_{\Omega_m} \quad (27)$$

where

$$\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u)|_{\Omega_s} = \|\sqrt{\tau}\mathbf{q}/\sqrt{\epsilon} - \sqrt{\tau}\sqrt{\epsilon}\nabla u\|_{0,\Omega_s}^2 + \| - \sqrt{\tau}/\bar{\kappa}_s\nabla\cdot\mathbf{q} + \sqrt{\tau}\bar{\kappa}_s u\|_{0,\Omega_s}^2 \quad (28)$$

and

$$\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u)|_{\Omega_m} = \|\mathbf{q}/\sqrt{\epsilon} - \sqrt{\epsilon}\nabla u\|_{0,\Omega_m}^2 + \|\nabla\cdot q\|_{0,\Omega_m}^2. \quad (29)$$

Integration by parts shows that

$$\int_{\Omega_m} \nabla \cdot \mathbf{q}u + \int_{\Omega_m} \mathbf{q} \cdot \nabla u - \int_{\Gamma} u\mathbf{q} \cdot \mathbf{n}_m = 0, \tag{30}$$

where $\mathbf{n}_m$ is the unit normal at Γ, pointing from the solute region into the solvent region. Applying this result to (29), we obtain

$$\begin{aligned}
\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u)|_{\Omega_m} &= \|\mathbf{q}/\sqrt{\epsilon}\|^2_{0,\Omega_m} + \|\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_m} \\
&\quad - 2\int_{\Omega_m} \mathbf{q} \cdot \nabla u + \|\nabla \cdot \mathbf{q}\|^2_{0,\Omega_m} + 2\tau \int_{\Omega_m} \nabla \cdot qu \\
&\quad + 2\tau \int_{\Omega_m} q \cdot \nabla u - 2\tau \int_{\Gamma} u\mathbf{q} \cdot \mathbf{n}_m + \tau^2\|u\|^2_{0,\Omega_m} - \tau^2\|u\|^2_{0,\Omega_m} \\
&= \|\mathbf{q}/\sqrt{\epsilon} + (\tau-1)\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_m} + \|\nabla \cdot \mathbf{q} + \tau u\|^2_{0,\Omega_m} \\
&\quad - \tau^2\|u\|^2_{0,\Omega_m} + (2\tau-\tau^2)\|\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_m} - 2\tau \int_{\Gamma} u\mathbf{q} \cdot \mathbf{n}_m.
\end{aligned} \tag{31}$$

Similarly, using integration by parts on eq. (28) yields

$$\begin{aligned}
&\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u)|_{\Omega_s} \\
&\quad = \|\sqrt{\tau}q/\sqrt{\epsilon}\|^2_{0,\Omega_s} + \|\sqrt{\tau}\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_s} - 2\tau \int_{\Omega_s} q \cdot \nabla u \\
&\quad\quad + \|\sqrt{\tau}\nabla \cdot \mathbf{q}/\bar{\kappa}_s\|^2_{0,\Omega_s} + \|\sqrt{\tau}\bar{\kappa}_s u\|^2_{0,\Omega_s} - 2\tau \int_{\Omega_s} \nabla \cdot qu \\
&\quad = \|\sqrt{(\tau}/\bar{\kappa}_s)\nabla \cdot \mathbf{q}\|^2_{0,\Omega_s} + \|\sqrt{\tau}q/\sqrt{\epsilon}\|^2_{0,\Omega_s} + \|\sqrt{\tau}\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_s} \\
&\quad\quad + \|\sqrt{\tau}\bar{\kappa}_s u\|^2_{0,\Omega_s} - 2\tau \int_{\Gamma} u\mathbf{q} \cdot \mathbf{n}_s,
\end{aligned} \tag{32}$$

where $\mathbf{n}_s = -\mathbf{n}_m$ is the unit normal along Γ, pointing from the solvent domain into the solute.

Using the Poincaré-Friedrichs inequality, we can assume

$$\|u\|^2_{0,\Omega} \le \lambda\|\nabla u\|^2_{0,\Omega}, \quad \text{with} \quad \lambda > 1. \tag{33}$$

From eqs. (27), (31), (32), (33) and choosing $\tau = \frac{1}{2\lambda} < 1$ we have

$$\begin{aligned}
\hat{\mathcal{F}}(\mathbf{q},u;\mathbf{q},u) &= \|\sqrt{\tau}q/\sqrt{\epsilon}\|^2_{0,\Omega_s} + \|\sqrt{\tau}\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_s} \\
&+ \|\sqrt{\tau}\nabla \cdot \mathbf{q}/\bar{\kappa}_s\|^2_{0,\Omega_s} + \|\sqrt{\tau}\bar{\kappa}_s u\|^2_{0,\Omega_s} + \|\mathbf{q}/\sqrt{\epsilon} + (\tau-1)\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_m} \\
&\quad + \|\nabla \cdot \mathbf{q} + \tau u\|^2_{0,\Omega_m} - \tau^2\|u\|^2_{0,\Omega_m} + (2\tau-\tau^2)\|\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_m} \\
&\quad \ge \|\sqrt{\tau}\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_s} + (2\tau-\tau^2)\|\sqrt{\epsilon}\nabla u\|^2_{0,\Omega_m} - \tau^2\|u\|^2_{0,\Omega_m} \\
&\quad\quad \ge \tau\|\sqrt{\epsilon}\nabla u\|^2_{0,\Omega} - \tau^2\|u\|^2_{0,\Omega} \ge (\tau - \lambda\tau^2)\|\sqrt{\epsilon}\nabla u\|^2_{0,\Omega} \\
&\quad\quad\quad = \frac{1}{4\lambda}\|\sqrt{\epsilon}\nabla u\|^2_{0,\Omega} \ge \alpha_3\|\nabla u\|^2_{0,\Omega},
\end{aligned} \tag{34}$$

where $\alpha_3 = \frac{1}{4\lambda}\min(\epsilon_m, \epsilon_s)$.

Now from eq. (20), we get

$$\mathcal{F}(\mathbf{q},u;\mathbf{q},u) \ge \alpha_4\|\nabla u\|^2_{0,\Omega}, \tag{35}$$

where $\alpha_4 = \alpha_1\alpha_3$. From the Poincaré-Friedrichs inequality (33), we find

$$\mathcal{F}(\mathbf{q},u;\mathbf{q},u) \ge \alpha_5\|u\|^2_{0,\Omega}. \tag{36}$$

Moreover,

$$\begin{aligned}
\|\mathbf{q}/\epsilon\|^2_{0,\Omega} &\le 2(\|\mathbf{q}/\epsilon - \nabla u\|^2_{0,\Omega} + \|\nabla u\|^2_{0,\Omega}) \\
&\le 2\left(1 + \frac{1}{\alpha_4}\right)\mathcal{F}(\mathbf{q},u;\mathbf{q},u),
\end{aligned}$$

and hence $\mathcal{F}(\mathbf{q},u;\mathbf{q},u) \ge \alpha_6\|\mathbf{q}\|^2_{0,\Omega}$ for $\alpha_6 = \alpha_4[2(1+\alpha_4)\max(\epsilon_s,\epsilon_m)]^{-1}$. Similarly,

$$\begin{aligned}
\|\nabla \cdot \mathbf{q}\|^2_{0,\Omega} &\le 2(\|\nabla \cdot \mathbf{q} - \bar{\kappa}^2 u\|^2_{0,\Omega} + \|\bar{\kappa}^2 u\|^2_{0,\Omega}) \\
&\le 2(1 + \bar{\kappa}_s^4\alpha_5)\mathcal{F}(\mathbf{q},u;\mathbf{q},u),
\end{aligned} \tag{37}$$

and thus $\mathcal{F}(\mathbf{q},u;\mathbf{q},u) \ge \alpha_7\|\nabla \cdot \mathbf{q}\|^2_{0,\Omega}$ for $\alpha_7 = [2(1+\bar{\kappa}_s^4\alpha_5)]^{-1}$.

Taking $\gamma_2 = \min(\alpha_4, \alpha_5, \alpha_6, \alpha_7)$ completes the proof. ∎

The FOSLS functional (15) is $H(\text{div}) \times H^1$ equivalent. In some FOSLS formulations, a curl term of the form $\nabla \times (\mathbf{q}/\epsilon) = 0$ is added to problem formulation (e.g., ref. 39), yielding a $H^1 \times H(\text{div}) \cap H(curl)$ equivalent FOSLS functional. The extra constraint is motivated by $\tilde{\mathbf{q}} = \epsilon\nabla u$, which implies $\nabla \times \tilde{\mathbf{q}}/\epsilon = 0$ (c.f., ref. 47, Theorem 2.9). However, for our case, we cannot take the curl of $\mathbf{q}/\epsilon$. This follows from our definition of $\mathbf{q} = \epsilon\nabla u + (\epsilon - \epsilon_m)\nabla u_c$; the curl of $\mathbf{q}/\epsilon$ is undefined at the interface. Hence we do not add the curl term to the formulation.

Traditionally, developing an effective error estimator for use in local adaptive refinement is challenging. Error estimators based on the Galerkin method are not immediately obvious from the problem formulation and local error bounds for the PBE can be complicated to derive.[3] In contrast, the FOSLS framework directly provides a natural error indicator through the functional. The local value of FOSLS functional is an *a posteriori* lower error bound, and, under some restrictions on mesh refinement, the bound can be shown to be a sharp theoretical error estimate.[48] We exploit this fact and build an adaptive refinement scheme based on the value of the FOSLS functional.

Let $G_\tau(\mathbf{q},u;u_c)$ be the value of the FOSLS functional (15) restricted to element τ. Note that if S is the set of elements comprising the mesh, then

$$G(\mathbf{q},u;u_c) = \sum_{\tau \in S} G_\tau(\mathbf{q},u;u_c).$$

Let $\mu_\tau = \sqrt{G_\tau(\mathbf{q},u;u_c)}$ and $\mu_{\max} = \max_{\tau \in S}\mu_\tau$. We mark simplex τ for refinement if $\mu_\tau \ge \gamma\mu_{\max}$, where $\gamma \in (0,1)$.

Our strategy is relatively straightforward, yet more advanced marking strategies based on the "solvation free-energy"[49] and FOSLS[48,50] functionals have been proposed in the literature. However, in our numerical experiments, we did not find a significant difference in performance when the marking strategy is varied for our problem. When compared on the same mesh, FOSLS requires more memory and CPU time than the standard second-order Galerkin method. However, the meshes produced by the corresponding adaptive refinement schemes are different, and the FOSLS approach is often able to achieve a more accurate solution with less refinement. As a result, the FOSLS approach is often more efficient than a standard second-order Galerkin method. The effectiveness of our scheme is highlighted in the following section.

Numerical Experiments

We use a tetrahedral mesh of Ω with globally continuous piecewise linear finite functions (P1 elements) and implement our finite element method and mesh refinement in FETK.[42] The meshes are generated using the Geometry-preserving Adaptive Mesher (GAMer), which is designed to produce simplicial meshes of molecular volumes and interfaces.[51] As a result, the solvent domain has a spherical outer boundary and the mesh is conforming at the interface of the solvent and molecule regions.

For the first four numerical experiments, we choose $\epsilon_m = 1$, $\epsilon_s = 78$, and $\bar{\kappa}_s = 0.918168$, which corresponds to a typical ionic strength of 0.1M. In these experiments, we solve for the regularized potential and strongly impose boundary conditions. The experiments are performed on the Born ion, Fasciculin 1, methanol, and a simple dipole. Let $\mathbf{q}^h$ and u^h be our finite-element solution, and $\mathbf{q}$ and u the true solution. We verify convergence to the solution by monitoring the square-root of FOSLS functional, $G(\mathbf{q}^h, u^h; u_c)^{\frac{1}{2}}$, as the FOSLS functional measures the error in the norm induced by G: $G(\mathbf{q}^h, u^h; u_c)^{\frac{1}{2}} = G(\mathbf{q}^h - \mathbf{q}, u^h - u; 0)^{\frac{1}{2}}$. Therefore, convergence of the FOSLS functional to zero implies convergence of our finite element solution to the true solution. We use uniform octal refinement and adaptive refinement to test the effectiveness of your method, with adaptive refinement being carried out by longest edge bisection. As $G(.,.,0)^{\frac{1}{2}}$ is equivalent to $H(\mathrm{div}) \times H^1$ norm, a standard finite-element error estimate implies optimal convergence rate to be $O(h)$ using uniform refinement with piecewise linear basis functions.[37] This optimal estimate assumes the problem to be H^2 regular. The convergence rate degrades as the solution becomes less smooth. We examine this scenario (dipole) and show that we still recover optimal convergence using adaptive refinement. In the following results, we refer to $G(\mathbf{q}^h, u^h; u_c)^{\frac{1}{2}}$ as the FOSLS norm and plot convergence rates normalized by the largest value.

Finally, to validate the solutions generated by our implementation, we compute the solvation free energy of transcription factor PML (PDB code 1BOR). We compare the computed value with values found in the literature.

Born Ion

Because of the complex geometries associated with molecules, there are few analytical solutions to the PBE or linearized PBE; however, it is possible to find an expression for the potential of a spherical ion in a solvent.[52] This system is referred to as the Born ion after its author Max Born.[53] The domain is consists of a spherical solute of radius R with a single point charge Q_1 at its center. The solute is surrounded by an unbounded solvent, Ω_s, as depicted in Figure 2a.

Writing the linear regularized PBE in spherical coordinates yields

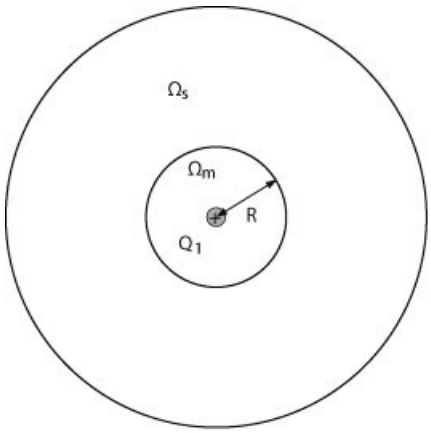
$$-\frac{1}{r^2}\frac{d}{dr}\left(\epsilon(r)r^2\frac{d}{dr}u(r)\right) + \bar{\kappa}^2(r)u(r) = -\bar{\kappa}^2(r)u_c(r), \quad r \neq R,$$

$$\left[\left[\epsilon(r)\frac{d}{dr}u(r)\right]\right]_\Gamma = (\epsilon_m - \epsilon_s)\frac{d}{dr}u_c(r), \quad r = R$$

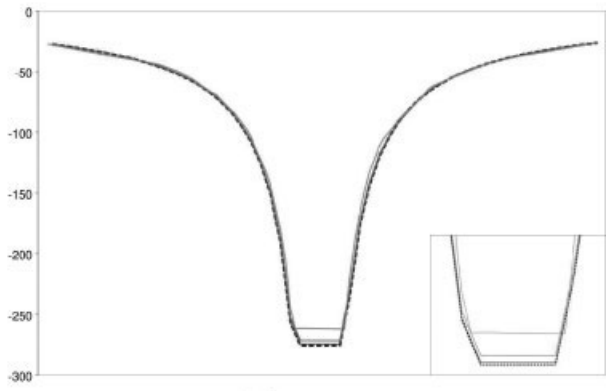
$$u(\infty) = 0$$

where $\omega = \bar{\kappa}_s/\sqrt{\epsilon_s}$. Following [52], we obtain the analytic solution

$$u(r) = \begin{cases} C_1 \exp[-\omega(r-R)]/r - C_2/r, & R \leq r, \\ (C_1 - C_2)/R, & 0 \leq r < R, \end{cases}$$



(a) Domain



(b) Convergence

Figure 2. Born ion.

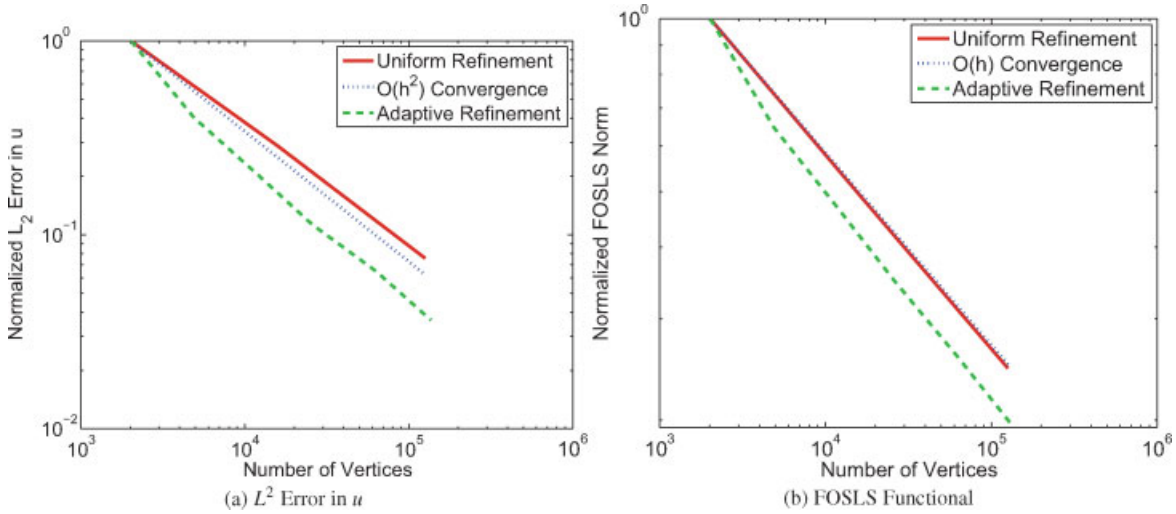


Figure 3. Convergence rates of born ion. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

where

$$C_1 = \frac{e_c Q_1}{k_B T} \frac{1}{(1+\omega R)\epsilon_s}, \quad \text{and} \quad C_2 = \frac{e_c Q_1}{k_B T} \frac{1}{\epsilon_m}.$$

Figure 3a displays the convergence of the reaction potential u in the L^2 norm, where the normalized L^2 error is plotted as a function of N, the number of points in the mesh. In three-dimensions, we observe a convergence rate of nearly $O(h^2)$ for uniform refinement, which corresponds to $O(N^{-2/3})$. On the other hand, for adaptive refinement, we observe a slightly better convergence rate. Figure 3b displays the FOSLS functional residual as the mesh is refined. In three-dimensions, a convergence rate of $O(h)$ corresponds to $O(N^{-1/3})$. We see that the FOSLS functional decreases nearly linear in h. During refinement, we ensure that new points on the solute/solvent interface lie on the analytically determined spherical boundary of the interface. As an example of convergence, in Figure 2b we display a slice of the true solution, a numerical solution on the initial mesh, and numerical solutions after two successive steps of uniform mesh refinement.

Fasciculin 1

The Born ion is a useful test case as the analytical solution is known; however, it is not a realistic simulation. To study the effectiveness of the FOSLS formulation on a realistic protein, we compute the regularized potential of Fasciculin 1 (1FAS in the Protein Data Bank) in an implicit solvent. 1FAS is a neurotoxin found in green mamba venom.[54] The dynamics and electrostatics of the Fasciculin 2 variant of this protein in its role as an acetylcholinesterase inhibitor have been studied in [55] and [56], where the electrostatics are argued to be important to its function. In our experiments, we use the description of the molecule specified in the PDB file from the Protein Data Bank and strip off water molecules using VMD.[57] The molecule region is not perfectly spherical, and we do not expect the solution to be symmetric as we did in the case of Born ion. It is assumed that the initial mesh defines the solute and solvent regions so that the solute/solvent interface in this case is polygonal and defined by the initial mesh. Consequently, refinement adds points to the polygonal interface. Although the analytical solution for Fasciculin 1 is not known, we are able to monitor the convergence of FOSLS functional.

Figure 4 shows the normalized convergence rate of FOSLS functional. Both uniform and adaptive refinement perform well, the convergence rate is better than $O(h)$ for both cases. Figure 5 depicts adaptive refinement around the Fasciculin molecule. The adaptive scheme refines aggressively around the areas where the solution is changing sharply.

Methanol

We examine our method in the more challenging setting of a methanol molecule, obtained from the APBS software package.[11]

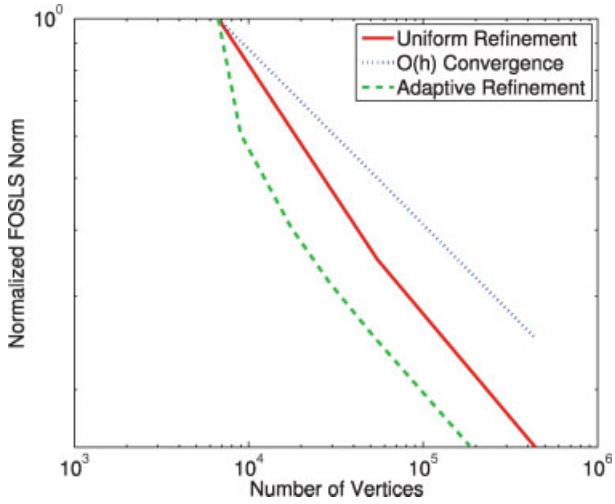
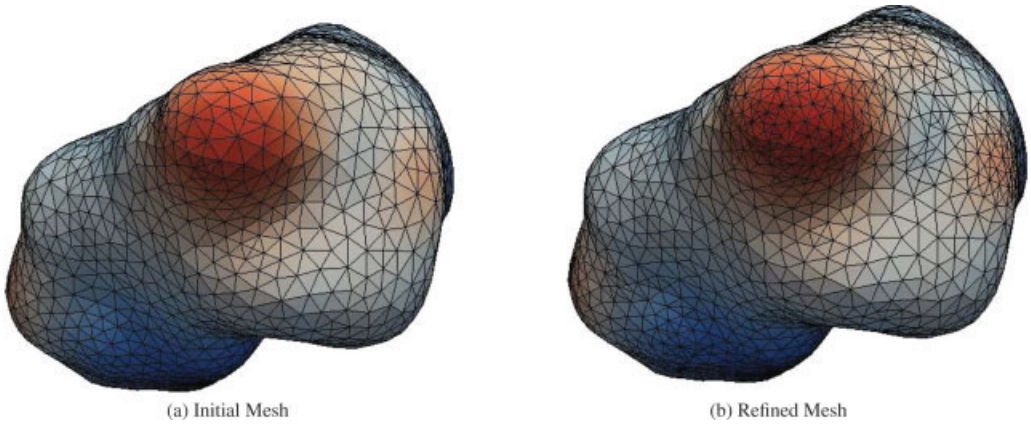
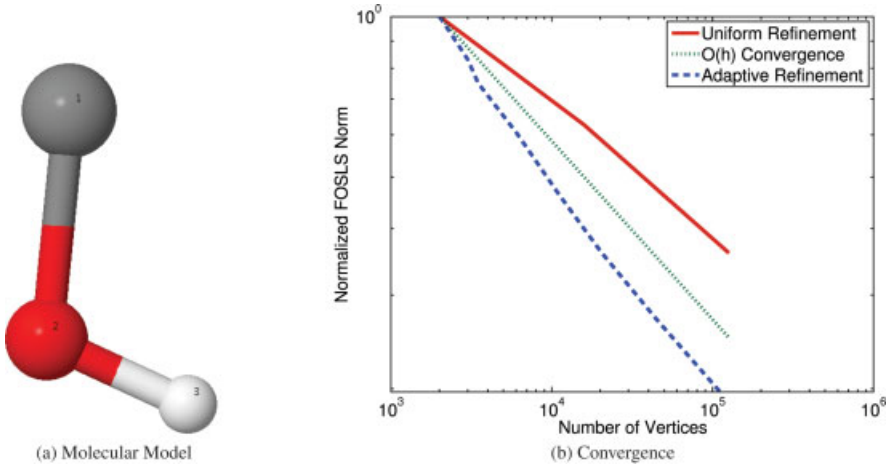


Figure 4. Fasciculin 1. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]



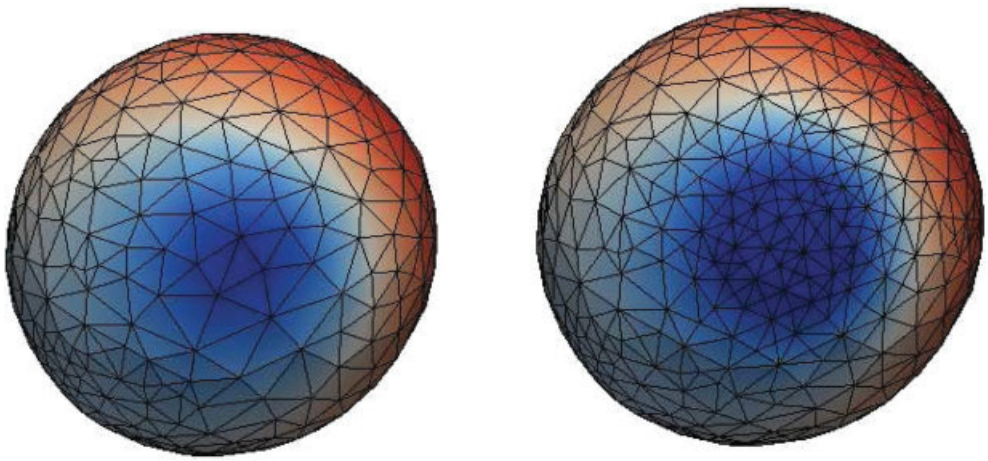
(a) Initial Mesh (b) Refined Mesh

Figure 5. Adaptive refinement around the solute/solvent interface of Fasciculin 1.



(a) Molecular Model (b) Convergence

Figure 6. Methanol.



(a) Initial Mesh (b) Refined Mesh

Figure 7. Adaptive refinement around the solute/solvent interface of methanol.

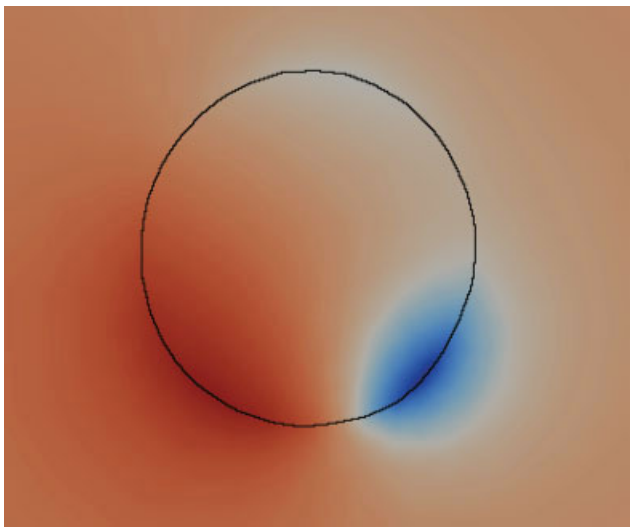


Figure 8. Methanol: Solution around the interface.

The model consists of three charged spheres representing charge groups: CH_3 and H with positive charges of 0.27 and 0.43, respectively, and an O atom with a negative charge of 0.7. The net charge on the molecule is zero. Figure 6a displays the methanol molecule.

We assume again that the initial mesh properly defines the solute and solvent regions. Figure 6b displays the FOSLS functional as a function of the number of vertices in the mesh. We see from the plot that the FOSLS functional does decrease, but the convergence is slightly less than $O(h)$. On the other hand, adaptive refinement is ideal for this problem as the solution varies sharply across the interface, indicating areas where local refinement is useful. As Figure 6b shows adaptive refinement yields slightly better than $O(h)$ convergence. The performance of adaptive refinement is shown in Figures 7 and 8, where the regularized electrostatic potential around the interface is displayed. Figure 7 shows the initial mesh and an adaptively refined mesh. Figure 8 displays a slice of the regularized solution, which highlights the areas in which the solution changes rapidly and also that the solution is not symmetric.

Dipole

In this section we illustrate the performance of our scheme on a simple dipole, as depicted in Figure 9. The linearized PBE for ions inside a spherical molecular region has been studied in.[58] For our experiment the domain consists of a spherical molecular region of radius 2 units, with two equal, but opposite unit charges, q^+ and q^-, inside. The charges are placed on opposite sides of the x-axis, each at distance d from the origin (see Fig. 9). As d is increased, the charges move closer to the interface, the solution becomes less well-behaved, developing a sharp gradient at the interface. Uniform refinement does not efficiently resolve the solution in this scenario. However, adaptive refinement is able to refine locally around the simplices at the interface, and gives a significantly better convergence rate than uniform refinement as shown in Figure 10. In particular, the rate of convergence for adaptive refinement is nearly insensitive to changes in the parameter d.

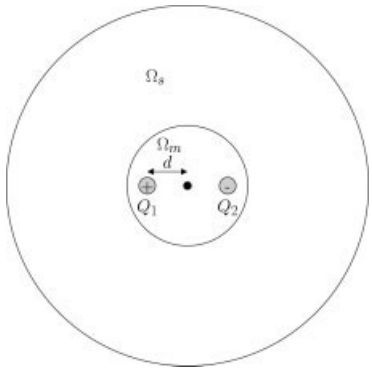


Figure 9. Domain for a simple dipole.

1BOR

Finally, we compute the electrostatic solvation free energy of transcription factor PML (PDB code 1BOR), and compare our value with the results in ref. 59, where they choose $\epsilon_m = 1$, $\epsilon_s = 80$. The electrostatic free energy of solvation is defined by[59]

$$\triangle G_{\text{sol}} = \frac{1}{2}\sum_{j=1}^{n_s} Q_j(\phi(x_j) - \phi_{\text{homo}}(x_j)), \tag{38}$$

where ϕ_{homo} is the solution of eq. (2a) in homogenous environment, thats is $\epsilon_m = \epsilon_s = 1$. In terms of the regularized potential u, the solvation free energy can be computed as,

$$\triangle G_{\text{sol}} = \frac{1}{2}\frac{k_B T}{e_c}\sum_{j=1}^{n_s} Q_j u(x_j) \tag{39}$$

On a mesh with 131,086 vertices, we compute the free energy of solvation equal to -792.577 kcal/mol, which compares well with the value of -853.7 kcal/mol computed from the MIBPB-III method in ref. 59. The free energy of solvation is sensitive to the geometry of the protein surface. We use GAMer to define this interface geometry, and hence our result does not exactly match up with,[59] who surface.

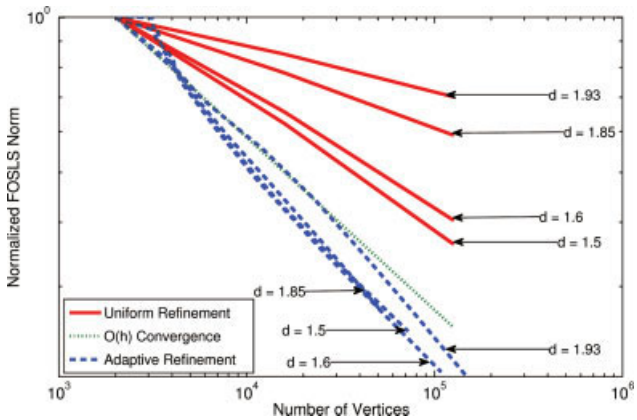


Figure 10. FOSLS functional convergence rates for a simple dipole.

Conclusion

The interface jump condition in (11) presents a challenge to design a single-domain FOSLS approach. We overcome this difficulty with a choice of a vector parameter **q** that results in a consistent and well-posed first-order system. The approach is also useful for solving the nonlinear equation using a Newton-FOSLS method,[61] as each step of Newton's method will effectively involve solution of a linearized Poisson-Boltzmann equation. In this article, we show that the resulting FOSLS functional defines a norm equivalent to the norm on $H^1 \times H(\mathrm{div})$, yet can be used in an existing finite element framework that uses more standard piecewise continuous elements in H^1.

We offer numerical evidence in support of this approach and test the methodology on several problems. We observe that adaptive refinement based on the FOSLS functional scheme yields a faster convergence rate than uniform refinement, and that this effect is more pronounced for solutions that are more sharply varying.

References

1. Baker, N. A.; Bashford, D.; Case, D. A. In: New Algorithms for Macromolecular Simulation, Vol. 49 of Lecture Notes in Computational Science and Engineering; Leimkuhler, B.; Chipot, C.; Elber, R.; Laaksonen, A.; Mark, A.; Schlick, T.; Schutte, C.; Skeel, R. Eds.; Springer-Verlag, Berlin, 2006, pp. 263–295.
2. Bashford, D.; Case, D. A. Annu Rev Phys Chem 2000, 51, 129.
3. Chen, L.; Holst, M. J.; Xu, J. SIAM J Numer Anal 2007, 45, 2298.
4. Fogolari, F.; Zuccato, P.; Esposito, G.; Viglino, P. Biophys J 1999, 76, 1.
5. Fogolari, F.; Brigo, A.; Molinari, H. J Mol Recognit 2002, 15, 377.
6. Klapper, I.; Hagstrom, R.; Fine, R.; Sharp, K.; Honig, B. Proteins 1986, 1, 47.
7. Warwicker, J.; Watson, H. C. J Mol Biol 1982, 157, 671.
8. Gilson, M. K.; Sharp, K. A.; Honig, B. H. J Comput Chem 1987, 9, 327.
9. Davis, M. E.; McCammon, J. A. J Comput Chem 1989, 10, 386.
10. Holst, M. J.; Saied, F. J Comput Chem 1995, 16, 337.
11. Baker, N. A.; Sept, D.; Joseph, S.; Holst, M. J.; McCammon, J. A. Proc Natl Acad Sci USA 2001, 98, 10037.
12. Luty, B. A.; Davis, M. E.; McCammon, J. A. J Comput Chem 1992, 13, 1114.
13. Bashford, D. In Scientific Computing in Object-Oriented Parallel Environments, Vol. 1343 of Lecture Notes in Computer Science; Springer, Berlin, 1997, pp. 233–240.
14. Rocchia, W.; Alexov, E.; Honig, B. J Phys Chem B 2001, 105, 6507.
15. Zhou, Y. C.; Wei, G. W. J Comput Phys 2006, 219, 228.
16. Orttung, W. H. Ann N Y Acad Sci 1977, 303, 22.
17. Cortis, C. M.; Friesner, R. A. J Comput Chem 1997, 18, 1591.
18. Holst, M. J.; Baker, N. A.; Wang, F. J Comput Chem 2000, 21, 1319.
19. Baker, N. A.; Holst, M. J.; Wang, F. J Comput Chem 2000, 21, 1343.
20. Shestakov, A. I.; Milovich, J. L.; Noy, A. J Colloid Interface Sci 2002, 247, 62.
21. Xie, D.; Zhou, S. BIT Numer Math 2007, 47, 853.
22. Wenbin, C.; Yifan, S.; Qing, X. Appl Math Comput 2005, 164, 11.
23. Miertus, S.; Scrocco, E.; Tomasi, J. Chem Phys 1981, 55, 117.
24. Zauhar, R.; Morgan, R. J Mol Biol 1985, 186, 815.
25. Rashin, A. A. J Chem Phys 1990, 94, 1725.
26. Yoon, B. J.; Lenhoff, A. M. J Comput Chem 1990, 11, 1080.
27. Juffer, A. H.; Botta, E. F.; van Keulen, B. A.; van der Ploeg, A.; Berendsen, H. J. J Comput Phys 1991, 97, 144.
28. Vorobjev, Y. N.; Grant, J. A.; Scheraga, H. A. J Amer Chem Soc 1992, 114, 3189.
29. Zhou, H.-X. Biophys J 1993, 65, 955.
30. Bharadwaj, R.; Windemuth, A.; Sridharan, S.; Honig, B.; Nicholls, A. J Comput Chem 1995, 16, 898.
31. Vorobjev, Y. N.; Scheraga, H. A. J Comput Chem 1997, 18, 569.
32. Liang, J.; Subramaniam, S. Biophys J 1997, 73, 1830.
33. Boschitsch, A. H.; Fenley, M. O.; Zhou, H.-X. J Phys Chem B 2002, 106, 2741.
34. Lu, B.; Zhang, D.; McCammon, J. A. J Chem Phys 2005, 122, 214102.
35. Kuo, S. S.; Altman, M. D.; Bardhan, J. P.; Tidor, B.; White, J. K. In ICCAD '02: Proceedings of the 2002 IEEE/ACM International Conference on Computer-Aided Design; ACM Press: New York, NY, 2002; pp. 466–473.
36. Lu, B.; Cheng, X.; Huang, J.; McCammon, J. A. Proc Natl Acad Sci USA 2006, 103, 19314.
37. Cai, Z.; Lazarov, R.; Manteuffel, T. A.; McCormick, S. F. SIAM J Numer Anal 1994, 31, 1785.
38. Cai, Z.; Manteuffel, T. A.; McCormick, S. F. SIAM J Numer Anal 1997, 34, 425.
39. Manteuffel, T. A.; McCormick, S. F.; Starke, G. In Seventh Copper Mountain Conference on Multigrid Methods, Vol. CP 3339; Melson, N. D.; Manteuffel, T. A.; McCormick, S. F.; Douglas, C. C., Eds.; NASA: Hampton, VA, 1996; pp. 535–550.
40. Berndt, M.; Manteuffel, T. A.; McCormick, S. F.; Starke, G. SIAM J Numer Anal 2005, 43, 386.
41. Berndt, M.; Manteuffel, T. A.; McCormick, S. F. SIAM J Numer Anal 2005, 43, 409.
42. Holst, M. Adv Comput Math 2001, 15, 139.
43. Bochev, P. B.; Gunzburger, M. D. SIAM Rev 1998, 40, 789.
44. Brezzi, F.; Fortin, M. Mixed and Hybrid Finite Element Methods, Vol. 15 of Springer Series in Computational Mathematics; Springer-Verlag, New York, 1991.
45. Song, Y.; Zhang, Y.; Shen, T.; Bajaj, C. L.; McCammon, J. A.; Baker, N. A. Biophys J 2001, 86, 2017.
46. Cao, Y.; Gunzburger, M. D. SIAM J Numer Anal 1998, 35, 393.
47. Girault, V.; Raviart, P.-A. Finite Element Methods for Navier-Stokes Equations, Vol. 5 of Springer Series in Computational Mathematics; Springer-Verlag, Berlin, 1986.
48. Berndt, M.; Manteuffel, T. A.; McCormick, S. F. Electron Trans Numer Anal 1997, 6, 35.
49. Cyr, E. C. Numerical Methods for Computing the Free-Energy of Coarse-Grained Molecular Systems, Phd Thesis; University of Illinois at Urbana-Champaign, Illinois, 2008.
50. Cai, Z.; Westphal, C. J Non-Newtonian Fluid Mech 2009, 159, 72.
51. Yu, Z.; Holst, M.; Cheng, Y.; McCammon, J. A. J Mol Graph Model 2008, 26, 1370.
52. Holst, M. J. Multilevel Methods for the Poisson-Boltzmann Equation, Phd Thesis; University of Illinois at Urbana-Champaign, 1994.
53. Born, M. Zeitschrift für Physik 1920, 1, 45.
54. le Du, M. H.; Marchot, P.; Bougis, P. E.; Fontecilla-Camps, J. C. J Biol Chem 1992, 267, 22122.
55. Radić, Z.; Kirchhoff, P. D.; Quinn, D. M.; McCammon, J. A.; Taylor, P. J Biol Chem 1997, 272, 23265.
56. Baker, N. A.; Helms, V.; McCammon, J. A. Proteins 1999, 36, 447.
57. Humphrey, W.; Dalke, A.; Schulten, K. J Mol Graph 1996, 14, 33.
58. Kirkwood, J. G. J Chem Phys 1934, 2, 351.
59. Geng, W.; Yu, S.; Wei, G. J Chem Phys 2007, 127, 114106.
60. Sanner, M. F.; Olson, A. J.; Spehner, J.-C. In SCG '95: Proceedings of the Eleventh Annual Symposium on Computational Geometry; ACM, New York, NY, 1995; pp. 406–407.
61. Codd, A. L.; Manteuffel, T. A.; McCormick, S. F. SIAM J Numer Anal 2003, 41, 2197.