Commun. Comput. Phys.
doi: 10.4208/cicp.OA-2017-0078

Vol. **23**, No. 2, pp. 520-539
February 2018

Accurate pKa Computation Using Matched Interface and Boundary (MIB) Method Based Poisson-Boltzmann Solver

Jingzhen Hu[1], Shan Zhao[2] and Weihua Geng[1]*

[1] *Department of Mathematics, Southern Methodist University, Dallas, TX 75248, USA.*
[2] *Department of Mathematics, University of Alabama, Tuscaloosa, AL 35487, USA.*

Received 29 March 2017; Accepted (in revised version) 16 May 2017

Abstract. The pK_a values are important quantities characterizing the ability of protein active sites to give up protons. pK_a can be measured using NMR by tracing chemical-shifts of some special atoms, which is however expensive and time-consuming. Alternatively, pK_a can be calculated numerically by electrostatic free energy changes subject to the protonation and deprotonation of titration sites. To this end, the Poisson-Boltzmann (PB) model is an effective approach for the electrostatics. However, numerically solving PB equation is challenging due to the jump conditions across the dielectric interfaces, irregular geometry of the molecular surface, and charge singularities. Our recently developed matched interface and boundary (MIB) method treats these challenges rigorously, resulting in a solid second order MIBPB solver. Since the MIBPB solver uses Green's function based regularization of charge singularities by decomposing the solution into a singular component and a regularized component, it is particularly efficient in treating the accuracy-sensitive, numerous, and complicated charge distributions from the pK_a calculation. Our numerical results demonstrate that accurate free energies and pK_a values are achieved at coarse grid rapidly. In addition, the resulting software, which pipelines the entire pK_a calculation procedure, is available to all potential users from the greater bioscience community.

AMS subject classifications: 92C40, 35J66

Key words: pKa, acid dissociation constant, Poisson-Boltzmann, finite difference, charge singularity.

1 Introduction

The acid dissociation constant K_a is a quantitative measure of the strength of an acid in solution, which is usually written as a quotient of the equilibrium concentrations as

*Corresponding author. *Email addresses:* wgeng@smu.edu (W. Geng), journeyh@smu.edu (J. Hu), szhao@ua.edu (S. Zhao)

$$K_a = \frac{[H^+][A^-]}{[HA]},$$

where [HA], [A^-], and [H^+] are concentrations of the acid, its conjugate, and proton in mol/L. The value $pK_a = -\log_{10} K_a$ is the co-logarithm of acid dissociation constant, which measures the tendency for a group to give up a proton. The smaller the value of pK_a, the more likely the acid is going to lose a proton, i.e. the stronger the acid is. Since proteins are chains of amino acids, the protonation or deprotonation of the titration sites (strongly polar amino acids) plays significant roles in binding affinities, enzymatic activities, and structural properties [1].

Since pK_a values are of significance to many biomolecular processes, their accurate measurement/calculation are practically important. For a short review, pK_a values can be measured by the following approaches.

Titration graph of acid-base reaction. The Henderson-Hasselbach equation shows $pH = pK_a + \log_{10}\left(\frac{[A^-]}{[HA]}\right)$. Thus pK_a=pH when [HA]=[A^-], which happens at half way of the titrant needed for reaching the equivalence point, i.e. the pH for which the site is 50% occupied. For a simple acid (e.g. the acetic acid), we can trace the pH during the titration process while adding base and then locate the pK_a on the titration curve.

NMR spectra. For proteins, however, pK_a of a titration site on a particular residue is hard to measure with acid-base reaction. NMR spectra in terms of chemical shifts are thus recorded as a function of pH [2,3]. The information of the chemical shift can indicate at which pH value the interested site is half-way protonated, and the corresponding pH is the desired pK_a.

Computer simulation. Since pK_a is associated with the thermodynamics of the acid dissociation [4], its values can also be predicted theoretically assisted with computer simulations. Various theoretical methods have been reported in literature including 1) Poisson-Boltzmann (PB) model [5–16], 2) Molecular Dynamics (MD) [17], 3) Monte Carlo (MC) method [18], 4) QM/MM (ab initio QM for the titratable residue and MM for the rest of the protein environment) [19], and 5) Empirical approaches [20,21].

We will focus on the PB model based pK_a computation, which assumes that the protonation or deprotonation asserts limited effect to the protein structure, and it is the titration states that bring the changes in electrostatic free energies. Under this assumption, one molecular structure is used for all titration states. The pK_a computation amounts to numerically solve PB equation for many times with different charge distributions, while repeatedly uses the same protein structure related information such as interface, mesh, elements, etc. This calls for an efficient and accurate PB solver.

However, numerically solving PB equation is challenging due to the jump conditions across the dielectric interface, irregular geometry of the molecular surface, and charge singularities. Although numerous PB solvers have been developed in the literature, the Matched Interface and Boundary method based Poisson-Boltzmann (MIBPB)

method [22–26] is chosen in this paper for pK_a computation, because the MIBPB method handles the interface jump conditions rigorously, treats the irregular geometry with delicately designed local interpolation, and regularizes the charge singularity using Green's function based decomposition. With these nice features, the MIBPB solver yields reliable electrostatic potential and free energy with a solid second order convergence. Comparing with other PB solvers, the MIBPB solver is particularly efficient in treating the accuracy-sensitive, abundant, and complicated charge distributions from the pK_a calculation. In this paper, the application of the well-established MIBPB solver to the pK_a computation will be reported for the first time in the literature. Our numerical results show that accurate electrostatic free energies are obtained even at coarse grid thus producing pK_a values at the combination of both efficiency and accuracy. In addition, we designed software to pipeline all procedures involved in pK_a calculation using MIBPB solver, enabling users to obtain pK_a values through one command line input.

The rest of the article is organized as follows. Section 2 is devoted to the theories and algorithms including the PB model, the MIB scheme, and algorithms for PB model based pK_a computation. Section 3 explains the procedure for using MIBPB solver to compute the pKa values, which serves an important part for disseminating the software released with this manuscript. Section 4 validates the MIBPB solver briefly and reports the numerical results in computing pKa for two selected proteins. This article ends with a brief conclusion summarizing the main points.

2 Theories and algorithms

In this section, we will first briefly describe the Poisson-Boltzmann (PB) model, and the scheme of the matched interface and boundary method (MIB) for solving PB equation. The treatment of charge singularity with the Green's function based two-component decomposition will be discussed. This section ends with theories and algorithms of PB model based pK_a computation.

2.1 The Poisson-Boltzmann (PB) model

The PB model is illustrated in Fig. 1(a). Consider a large domain Ω in $\mathbb{R}^3$ containing the solute protein. The domain Ω is divided by the molecule surface Γ into the molecule domain Ω^- with dielectric constant ϵ_1 and the solvent domain Ω^+ with dielectric constant ϵ_2, that is $\Omega = \Omega^- \bigcup \Omega^+$. Denote the boundary of Ω as $\partial\Omega$. Charges in Ω^- are partial charges assigned to the centers of atoms by using force field while charges in Ω^+ are mobile ions described by the Boltzmann distribution. For $\mathbf{r} \in \mathbb{R}^3$, applying Gauss's law to the charge distribution in both Ω^- and Ω^+ leads to the linear PB equation

$$-\nabla \cdot (\epsilon(\mathbf{r}) \nabla \phi(\mathbf{r})) + \bar{\kappa}^2(\mathbf{r}) \phi(\mathbf{r}) = \rho(\mathbf{r}) \tag{2.1}$$

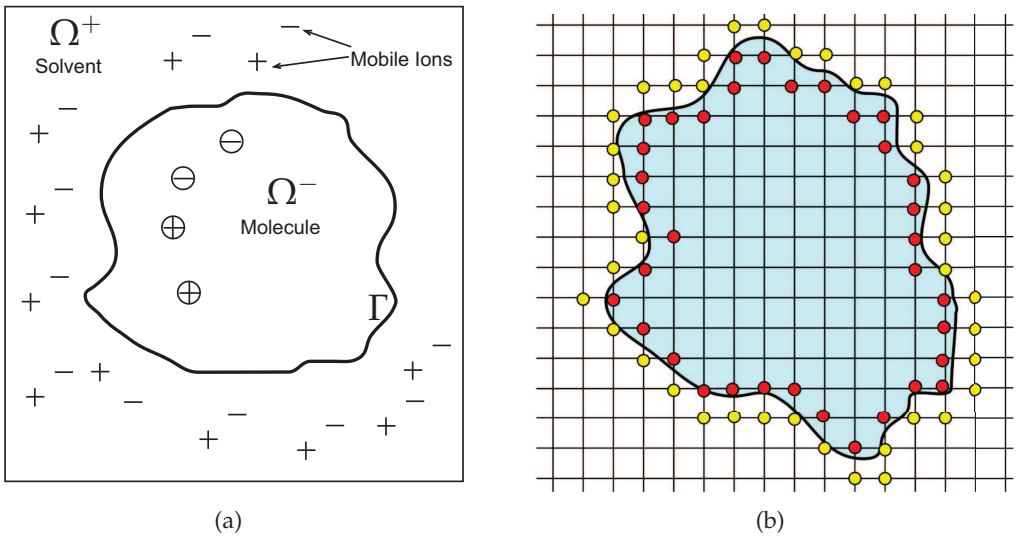


Figure 1: (a) The PB model; (b) The MIB scheme illustrated in the 2-D second order scheme with inside fictitious points in red and outside fictitious points in yellow.

subject to interface continuity for the potential ϕ and flux density $\epsilon\phi_{\mathbf{n}}$

$$[\phi]_\Gamma = 0 \quad \text{and} \quad [\epsilon\phi_{\mathbf{n}}]_\Gamma = 0 \quad \text{on } \Gamma, \tag{2.2}$$

where $\mathbf{n}$ is the outer normal direction of the interface Γ, $\phi_{\mathbf{n}} = \frac{\partial\phi}{\partial\mathbf{n}}$, and the notation $[f]_\Gamma = f^+ - f^-$ is the difference of the function f cross the interface Γ.

In Eqs. (2.1) and (2.2), ϵ is a piecewise function for the dielectric constants in Ω^- and Ω^+, and κ is the inverse Debye screening length measuring ionic strength, and its modified version $\bar{\kappa}$ is given as $\bar{\kappa}^2 = \epsilon_2\kappa^2$. The value of κ is nonzero in Ω^+ only. The reader can refer to [27,28] for more details about definition and units related to these coefficients. The source term ρ is the summation of the charge distribution in Ω^- using delta function for N_c partial charges located at $\mathbf{r}_i$ for $i = 1,\cdots,N_c$ as $\rho(\mathbf{r}) = 4\pi\sum_{i=1}^{N_c} q_i\delta(\mathbf{r}-\mathbf{r_i})$. Since the source $\rho(\mathbf{r})$ is the sum of the δ-functions, which brings singularity and numerically its interaction with the grid points is the source of so-called self-energies. We will present a new regularization approach to remove the singularity under the matched interface and boundary (MIB) framework [26].

2.2 Numerical methods for solving PB equation

Since the PB equation has analytic solution only for simple geometries for examples sphere and rod, in practice the equation is solved numerically. Numerical methods for solving PB equation fall into two classes, (1) grid-based methods that discretize the entire domain, e.g. [25,29–35], and (2) boundary integral methods that discretize the molecular

surface, e.g. [36–43]. The reader may see [44, 45] for comprehensive review for numerical methods in solving PB equations. The numerical solution to the PB equation suffers many challenges such as (1) the interfacial molecular surface is geometrically complex; (2) the biomolecule is represented by singular point charges; (3) the dielectric function and electric field are discontinuous across the molecular surface; (4) the nonlinearity appears when ionic strength is strong; (5) the domain is unbounded. In treating these numerical challenges, particularly (1)-(3), a finite-difference based interface method named matched interface and boundary (MIB) method stands out [22–25]. We next briefly introduce the scheme of MIBPB solver with particular focus on our newly developed two-component regularization scheme [26] as it is by far the most efficient approach under the MIB framework for treating the charge singularities.

2.3 The Matched Interface and Boundary method based Poisson-Boltzmann (MIBPB) solver

The MIB method is designed to capture discontinuities of the solutions and coefficients of the differential equations by using local high order interpolation assisted with the interface jump conditions. Based on the MIB method for general interface problems [22,46,47], the MIBPB solver is further improved to be able to maintain its designed 2nd order of convergence and accuracy in the presence of geometric singularities of molecular surfaces [23].

The essential idea of discretization in MIB scheme is the introduction of fictitious points as seen in Fig. 1(b). The yellow dots in outside domain are fictitious points extended from the inside domain while the red dots in inside domain are fictitious points extended from the outside domain. Each fictitious point has two values: one is the fictitious value as if it were a point extended from the opposite domain and the other is the actual value as the solution to the PB equation. The fictitious values can be solved by using interface jump conditions together with the PB equation. As a result, fictitious values are linear combination of the solution at nearby grid points and the interface jump conditions. When a 2nd order finite difference scheme is applied, the fictitious values will replace the actual values in the centered difference stencil, which eventually forms the linear algebraic matrix to be solved by Krylov iterative methods. The fictitious values based discretization guarantees the designed order of convergence at the presence of interface jump conditions [22]. Furthermore, when geometric singularities are present, the delicately designed local interpolation scheme can still maintain the designed order of accuracy [23].

A challenging issue in the MIBPB solver is the efficient treatment of charge singularities. To this end, we use the recently developed two-component singular charge decomposition scheme [26], which maintains the 2nd order accuracy while it is much more convenient in implementation compared with the previous three-component method [24]. To save space, we here only give a brief introduction to the singular charge decomposition scheme, which plays a critical role in the accurate and efficient computation of pK_a

values as described below.

We consider a decomposition of the electrostatic potential into a Coulomb component ϕ_C and a reaction field component ϕ_{RF} with $\phi=\phi_C+\phi_{\mathrm{RF}}$. The Coulomb potential ϕ_C satisfies the free space Poisson's equation with the singular charges as

$$\begin{cases} -\epsilon_1\Delta\phi_C(\mathbf{r})=\rho(\mathbf{r}), & \text{in } \mathbb{R}^3; \\ \phi_C(\mathbf{r})=0, & \text{as } |\mathbf{r}|\to\infty, \end{cases} \tag{2.3}$$

where $\rho=\sum_{i=1}^{N_c}4\pi q_i\delta(\mathbf{r}-\mathbf{r}_i)$. Here ϕ_C, which can be analytically given as

$$\phi_C(\mathbf{r})=G(\mathbf{r})=\sum_{i=1}^{N_c}\frac{q_i}{\epsilon_1|\mathbf{r}-\mathbf{r}_i|}, \tag{2.4}$$

is essentially the Green's function $G(\mathbf{r})$. We define a regularized potential $\tilde{\phi}$ as

$$\tilde{\phi}=\begin{cases} \phi_{\mathrm{RF}}^- & \text{in } \Omega^-, \\ \phi^+ & \text{in } \Omega^+. \end{cases} \tag{2.5}$$

The jump conditions for ϕ in Eq. (2.2) using the definition $\phi^-=\phi_C^-+\phi_{RF}^-$ in Ω^- can be written as

$$\phi^+=\phi_{RF}^-+\phi_C^-, \quad \epsilon_2\frac{\partial\phi^+}{\partial n}=\epsilon_1\frac{\partial\phi_{RF}^-}{\partial n}+\epsilon_1\frac{\partial\phi_C^-}{\partial n}, \quad \text{on } \Gamma. \tag{2.6}$$

Using $\tilde{\phi}^+=\phi^+$ in Ω^+ and $\tilde{\phi}^-=\phi_{\mathrm{RF}}^-$ in Ω^- as in Eq. (2.5), the regularized PB equation of $\tilde{\phi}$ with corresponding interface and boundary conditions is given as

$$-\nabla\cdot(\epsilon_1\nabla\tilde{\phi})=0 \qquad \text{in } \Omega^-, \tag{2.7}$$

$$-\nabla\cdot(\epsilon_2\nabla\tilde{\phi})+\bar{\kappa}^2\tilde{\phi}=0 \qquad \text{in } \Omega^+, \tag{2.8}$$

$$[\tilde{\phi}]_\Gamma=G \qquad \text{on } \Gamma, \tag{2.9}$$

$$\left[\epsilon\frac{\partial\tilde{\phi}}{\partial n}\right]_\Gamma=\epsilon_1\frac{\partial G}{\partial n} \qquad \text{on } \Gamma. \tag{2.10}$$

Note that $\tilde{\phi}$ actually satisfies the similar PB equation as in Eq. (2.1) but free of the singular source terms

$$-\nabla\cdot(\epsilon\nabla\tilde{\phi}(\mathbf{r}))+\bar{\kappa}^2\tilde{\phi}(\mathbf{r})=0, \quad \text{in } \Omega^-\cup\Omega^+. \tag{2.11}$$

Numerically, one just needs to solve one PB interface problem given in (2.7)-(2.10), then the original potential ϕ is recovered as $\phi^+=\tilde{\phi}^+$ in Ω^+ and $\phi^-=\tilde{\phi}^-+G$ in Ω^-, where Green's function G is analytically given as in Eq. (2.4). Because the singular charges are handled analytically, this two-component regularization method is well suited for the pK_{a} computation, in which various states are represented by different charge distributions.

2.4 PB model based pK_a computation

The solution to the PB model is the electrostatic potential on the grids, which can be used to further compute the electrostatic free energies. These energies at different titration states are the building blocks to compute the pK_a value.

2.4.1 Electrostatic free energy

As described by Sharp and Honig [48], the electrostatic free energy for a solvated biomolecule is calculated by

$$\Delta G_{\text{elec}} = \int_{\mathbb{R}^3} \left(\phi\rho + \Delta\Pi - \frac{1}{2}\epsilon|\mathbf{E}|^2 \right) d\mathbf{r}, \tag{2.12}$$

where ϕ is the electrostatic potential, ρ is the fixed charge density as a collection of point charges, $\Delta\Pi$ is the excess osmotic pressure of the mobile ion cloud, and $\frac{1}{2}\epsilon|\mathbf{E}|^2$ is the electrostatic stress.

In our numerical validation in the next section, we report the electrostatic free energy as

$$\Delta G_{\text{elec}} = E_{\text{coul}} + \frac{1}{2}\sum_{i=1}^{N_c} q_i \phi_{RF}(\mathbf{r}_i), \tag{2.13}$$

where ϕ_{RF} is the electrostatic potential inside molecule as seen in Eq. (2.5) and E_{coul} is the Coulomb energy using ϵ_1 as the dielectric constant. Here for simplification, we omit the energy components related to the mobile ion pressure and the electrostatic stress, whose contribution are relatively smaller but computationally more challenging as integrals of discontinuous functions. Details of evaluating the complete functional can be found in [48–50].

2.4.2 pK_a from thermodynamics

From thermodynamics (e.g. [4], pp. 123-127), we know an equilibrium constant is related to the standard Gibbs energy change for the reaction. Thus for an acid dissociation constant K_a, we have $\Delta G = -RT\ln K_a \approx (2.303RT)\text{p}K_a$, where $R = 8.31\text{J}/(\text{mol}\cdot\text{K})$ is the universal gas constant. Based on this, pK_a can be computed via electrostatic free energy [8, 9, 14, 16].

We can use the thermodynamics loop in Fig. 2 to compute the pK_a of a particular titration site in protein [1, 9]. Note in this derivation we use microscopic $k_B = R/N_A = 1.38\times10^{-23}\text{J/K}$ instead of the macroscopic R since we are dealing with an individual protein instead of molar concentration. Here the subscripts "p" and "s" represent protein and solvent environment respectively

$$\text{p}K_a(\text{protein}) = \text{p}K_a(\text{model}) + \frac{1}{2.303k_BT}[\Delta G_{\text{s,p}}(\text{A}) - \Delta G_{\text{s,p}}(\text{AH})] \tag{2.14}$$

$$= \text{p}K_a(\text{model}) + \frac{1}{2.303k_BT}[\Delta G^{\text{elec}}_{\text{s,p}}(\text{A}) - \Delta G^{\text{elec}}_{\text{s,p}}(\text{AH})] \tag{2.15}$$

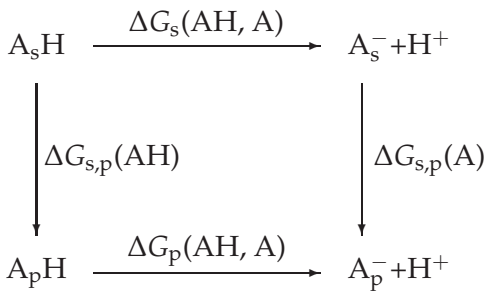


Figure 2: Thermodynamics circles involving acid dissociation in protein (p) and solvent (s); AH and A stand for protonated and unprotonated acids respectively.

$$= pK_a(\text{model}) + \frac{1}{2.303k_BT}[\Delta G_p^{\text{elec}}(\text{AH,A}) - \Delta G_s^{\text{elec}}(\text{AH,A})] \tag{2.16}$$

$$\begin{aligned} = pK_a(\text{model}) &+ \frac{1}{2.303k_BT}[\Delta G_{\text{Born}}(\text{protein}) - \Delta G_{\text{Born}}(\text{model}) \\ &+ \Delta G_{\text{back}}(\text{protein}) - \Delta G_{\text{back}}(\text{model}) + \Delta G_{\text{interact}}(\text{protein})] \end{aligned} \tag{2.17}$$

$$= pK_{\text{intr}} + \frac{1}{2.303k_BT}\Delta G_{\text{interact}}(\text{protein}). \tag{2.18}$$

In this derivation, Eq. (2.15) holds under the assumption that the non-polar effect from solvent to protein environment of unprotonated states cancels that of protonated states, leaving only the electrostatic effect. Eq. (2.16) is from the thermodynamics loop in Fig. 2. Bashford et al. [9] suggested to use Eq. (2.16) since $\Delta G_p(\text{AH, A})$ and $\Delta G_s(\text{AH, A})$ have small difference in structure (e.g. AH and A in protein vs AH and A in solvent) as opposed to Eq. (2.15) (e.g. AH in protein and solvent vs A in protein and solvent). This is also our choice. Eqs. (2.17) and (2.18) are derived using physical interpretation naturally and we next show this derivation is consistent with the energy change due to protonation. These derivations are originated from Bashford et al. [9] and we restate here with more details.

The energy associated with the shift pK_a at site i (the work of adding a charge q_i) can be computed as:

$$\Delta G_i = \underbrace{\Delta G_{\text{Born}} + \Delta G_{\text{back}}}_{\text{intrinsic-shift}} + \underbrace{\sum_j q_i q_j \Phi(r_i, r_j)}_{\text{interact}}, \tag{2.19}$$

with $\Phi(r_i, r_j)$ defined as potential produced at r_j by a unit charge at r_i for a solvated protein system (given solute-solvent surface Γ, dielectric constants ϵ_1 in solute and ϵ_2 in solvent and ionic screening constant κ). We can further decompose

$$\Phi(r_i, r_j) = 1/(\epsilon_1 |r_i - r_j|) + \Phi^*(r_i, r_j)$$

as the summation of Coulomb interaction and a correction term Φ^*. Physically, Φ^* is the potential generated from the induced charge on the dielectric interface (i.e. reaction

potential). In addition, $\Delta G_{\text{back}} = \sum_k q_i q_k \Phi(r_i, r_k)$ is the interaction with the background partial charges and $\Delta G_{\text{Born}} = (q_i^2/2)\Phi^*(r_i, r_i)$ is the interaction of a charge at r_i with a potential at r_i. Let's add two notes here as:

(1) $\Phi(r_i, r_j)$ can also be treated as the solution at r_j to the following single-charge (unit charge located at r_i) Poisson-Boltzmann equation:

$$-\varepsilon_1 \nabla^2 \phi(\mathbf{r}) = 4\pi\delta(\mathbf{r} - r_i), \quad \mathbf{r} \in \Omega^-, \tag{2.20a}$$

$$-\varepsilon_2 \nabla^2 \phi(\mathbf{r}) + \bar{\kappa}^2 \phi(\mathbf{r}) = 0, \quad \mathbf{r} \in \Omega^+. \tag{2.20b}$$

(2) For the "intrinsic-shift" part in Eq. (2.19), we can derive it from free energy differences before and after charging (e.g. the protonation of a charge at the $n+1$th location in addition to the n charged locations). For convenience, we define $\Phi_i(r_j) = \Phi(r_i, r_j)$ as the total potential at r_j from a unit charge at r_i, and $\Phi_i^*(r_j)$ is the corresponding reaction potential, both are involved in the energies with or without protonation as

$$G_{\text{deprot}} = \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\Phi_i^*(r_j) q_i q_j + \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1, j\neq i}^{n}\frac{q_i q_j}{\epsilon_1 |r_i - r_j|}, \tag{2.21}$$

$$G_{\text{prot}} = \frac{1}{2}\sum_{i=1}^{n+1}\sum_{j=1}^{n+1}\Phi_i^*(r_j) q_i q_j + \frac{1}{2}\sum_{i=1}^{n+1}\sum_{j=1, j\neq i}^{n+1}\frac{q_i q_j}{\epsilon_1 |r_i - r_j|}. \tag{2.22}$$

Note the first terms in Eqs. (2.21) and (2.22) are essentially the solvation free energies and we refer them as $G_{\text{deprot}}^{\text{solv}}$ and $G_{\text{deprot}}^{\text{solv}}$

$$\Delta G_{\text{prot}} = G_{\text{prot}} - G_{\text{deprot}} \tag{2.23}$$

$$= \underbrace{\sum_{i=1}^{n}\left(\Phi_i^*(r_{n+1}) q_i q_{n+1} + \frac{q_i q_{n+1}}{\epsilon_1 |r_i - r_{n+1}|}\right)}_{\Delta G_{\text{back}}} + \underbrace{\Phi_{n+1}^*(r_{n+1}) q_{n+1}^2}_{\Delta G_{\text{Born}}} \tag{2.24}$$

$$= G_{\text{prot}}^{\text{solv}} - G_{\text{deprot}}^{\text{solv}} + \sum_{i=1}^{n}\frac{q_i q_{n+1}}{\epsilon_1 |r_i - r_{n+1}|}. \tag{2.25}$$

Note Eq. (2.24) holds based on the symmetry of $\Phi_i^*(r_j)$, which is consistent with Eq. (2.19). In summary, the pK_a for a titration site is composed of the intrinsic pK_a and the site-site interaction as in Eq. (2.18). The free energy associated with intrinsic pK_a shift (away from the model pK_a) is given in Eq. (2.23) as the energy difference between protonation and deprotonation of the titration with only the background charges. The site-site interaction is computed as in the interaction term of Eq. (2.19). In the next section, we implement the pK_a computation based on Eqs. (2.18) and (2.19).

3 Implementation

3.1 pKa computing procedure

Following the discussion in the previous section, we use the programming language Python to write a wrapper to pipeline the pKa computation in four steps as 1) changing the protonation states on PDB and then on PQR files, with each PQR file representing a needed charge distribution in the pK_a computation, 2) calling MIBPB solver for solving electrostatics potential and free energies for all PQR files, 3) calculating the intrinsic pKa values, and 4) finally titrating the final pK_a values with energies including site-site interactions. The procedure implemented with the Python wrapper is given as the following.

Step 1: Prepare the protein structure and protonation states

In this step, we prepare the structure and charge distribution of the protein. The protein structure is obtained from the Protein Data Bank (PDB) (www.pdb.org). The charge distribution is then produced by PDB2PQR [51] with user chosen forcefields. Before calling the PDB2PQR program, the protonation states are configured by using different residue names as specified below in Table 1.

Table 1: The protonation states of titratable amino acids: each residue has a name for unprotonated state and a name for protonated states; total charge of the residue in a state is included in the parentheses.

standard resid.	**ARG**	**ASP**	**GLU**	**HIS**	**LYS**	**TYR**	**CYS**
unprot.(chg)	AR0 (+1)	ASP(-1)	GLU(-1)	HIE(0)	LYN(0)	TYM(-1)	CYM(-1)
protonated(chg)	ARG(+2)	ASH(0)	GLH(0)	HIP(+1)	LYS(+1)	TYR(0)	CYX(0)

For a protein with N_t titration sites, we will need $1+2N_t+\frac{1}{2}(N_t^2-N_t)$ charge distributions in the form of PQR files as specified below.

1. one PQR file with all titration sites unprotonated, keeping the background charge;
2. $2N_t$ PQR files having all titration sites unprotonated but one protonated with or without the background charge; those with background charges on are used for calculating the intrinsic pK_a;
3. $\frac{1}{2}(N_t^2-N_t)$ PQR files with ith and jth titration sites protonated only; all background charges are set to zero.

Step 2: Call MIBPB solver for solving electrostatics

This step calculates the electrostatic free energies for all titration states represented by different PQR files. Each PQR file with different charge distribution changes the RHS of the PB equation as in Eq. (2.1). The PB equation can then be accurately and conveniently solved using the MIBPB solver. In calling the MIBPB solver, the user can specify solver and PB model related parameters such as dielectric constants, ion concentration, mesh

size, etc. in the *usrdata.in* file. After this step, we received electrostatic free energies from all charge distributions. These energies will be used to calculate pK_a as explained in the next two steps. Our work follows the procedure as described in Ho's thesis [1].

Step 3: Compute the intrinsic pK_a

The intrinsic pK_a for the ith titratable site can be computed by Eq. (3.1) [1],

$$\mathrm{p}K_{a,i}^{intr} = \mathrm{p}K_{a,i}^{0} - \frac{1}{RT\ln 10}[\Delta G_{ele}(A_p \to A_pH) - \Delta G_{ele}(A_s \to A_sH)], \tag{3.1}$$

where the protein environment has only the fixed background charges, i.e. with all titration sites unprotonated. In this equation, $\Delta G_{ele}(A_p \to A_pH)$ is the difference of the free energy between protein with ith titration site protonated and protein with all titration unprotonated while $\Delta G_{ele}(A_s \to A_sH)$ is the difference of the free energy between protonated and unprotonated residues alone. Here $\mathrm{p}K_{a,i}^{0}$ is the model pK_a at $T = 298$K taken from [52] as in Table 2. For the related constants, R is the gas constant and T is the temperature in Kelvin. R is related to the Boltzmann constant, $k_B = 1.3806 \cdot 10^{-23}$J/K, and the Avogadro constant, $N_A = 6.02 \cdot 10^{23}$/mol, as

$$R = k_B \cdot N_A \approx 8.31\mathrm{J}/(\mathrm{mol}\cdot\mathrm{K}), \tag{3.2}$$

Since the energy calculated from MIBPB using the unit kcal/mol, we finally use the RT values as

$$RT \approx 8.31 \cdot 298\mathrm{J/mol} \approx 2.5\mathrm{kJ/mol} = (2.5/4.182)\mathrm{kcal/mol}, \tag{3.3}$$

thus the energy from MIBPB divided by $(2.5\ln 10/4.182 = 1.3765)$ kcal/mol leads to the unit of pK_a values.

Table 2: Model pK_a value for titration sites.

ARG	ASP	CYS	GLU	HIS	LYS	TYR
12.0	4.0	9.5	4.4	6.3	10.4	9.6

Step 4: Titrating final pK_a with energies including site-site interactions

Recall that pK_a of a titration site is defined as the pH value in which half of the site is protonated. In the context of computing pKa using electrostatic free energy under different titration states, we are looking for the pH which makes the Boltzmann average $<\theta_i, \mathrm{pH}>$ as in Eq. (3.4) to equal 0.5.

$$<\theta_i, \mathrm{pH}> \; = \frac{\sum_\theta \theta_i e^{-\triangle G(A \to A(\theta);\ \mathrm{pH})/RT}}{\sum_\theta e^{-\triangle G(A \to A(\theta);\ \mathrm{pH})/RT}}, \tag{3.4}$$

where $\theta \in \{0,1\}^{N_t}$ and $\theta_i \in \{0,1\}$ is the ith entry of θ. Note the number of states of θ increases exponentially (2^{N_t}) with number of titrating sites N_t, which is computationally

prohibitive. In this paper we use direct formulation for proteins with less than 20 titrating sites and we are working on statistical sampling approaches to handle cases with larger N_t for future work.

In Eq. (3.4), the energy $\triangle G(\mathrm{A}\to\mathrm{A}(\theta);\mathrm{pH})$ is evaluated as

$$\triangle G(\mathrm{A}\to\mathrm{A}(\theta);\mathrm{pH})=-RT\ln 10\sum_i\theta_i(\mathrm{p}K_{\mathrm{a},i}^{intr}-\mathrm{pH})+\frac{1}{2}\sum_i\theta_i\sum_{j\neq i}\theta_j\triangle G_{ij}, \tag{3.5}$$

where $\mathrm{p}K_{\mathrm{a},i}^{intr}$ is the intrinsic $\mathrm{p}K_\mathrm{a}$ for the ith titration site as calculated in step 3 by Eq. (3.1). $\triangle G_{ij}$ as calculated in Eq. (3.6) is the site-site interaction energy (the free energy of the protein having the ith site protonated for producing electrostatic potential and the jth titration site protonated for producing energy without the background charge) computed in step 2 as well.

$$\triangle G_{ij}=t_i^TWt_j=\frac{1}{2}(t_i+t_j)^TW(t_i+t_j)-\frac{1}{2}t_i^TWt_i-\frac{1}{2}t_j^TWt_j. \tag{3.6}$$

The second equality holds under the assumption that W is symmetric.

3.2 Software dissemination and user guide

With the publication of this manuscript, the python wrapper (*wrapper_pka.py*) and the binary MIBPB solver (*mibpb3.exe* on multiple platforms such as MacOS, Linux/Unix, and Windows) can be found on the author's website sponsored by Southern Methodist University. Since part of the MIBPB solver source code is copyrighted to Michigan State University, please contact Dr. Guowei Wei if interested. The user also needs the MSMS software [53] for molecular surface generation and we include its binary versions as well. In addition, the users need to install PDB2PQR [51].

To compute pKa, the user specifies PB model and MIBPB related parameters (dielectric constants, ion concentration, mesh size, density of molecular surface triangulation, boundary conditions, method of charge regularization, etc.) in *usrdata.in* file. On a computer with Python and Fortran compilers installed and target protein specified with its four-digit PDB ID, the user runs the program by simply typing:

```
python wrapper_pka.py PDBID
```

The wrapper will automatically download PDB file from the protein data bank, identify all titration sites, call MIBPB solver for electrostatics, and return the computed $\mathrm{p}K_\mathrm{a}$ values.

4 Results

In this section, we first validate the accuracy of MIBPB solver, which justifies our choice of mesh size $h=1.0$ for all PB related calculations. Following that, we calculate the $\mathrm{p}K_\mathrm{a}$ values for two selected proteins, which are compared with the data from experiments.

4.1 Validation of MIBPB solver

The extensive validation of new MIBPB method in solving PB equation can be found in [26]. Here we provide results of solving PB equation on bovine pancreatic trypsin inhibitor (BPTI, PDB ID: 4pti) [54] and turkey ovomucoid third domain (OMTKY3, PDB ID: 2ovo) [55] at various finite difference mesh sizes. For simplicity, we refer these two proteins with their PDB IDs 4pti and 2ovo hereafter. The two proteins are also the targets for our pK_a computation. The resulting free energies and errors at different mesh sizes and solute dielectric constants are given in Table 3.

Table 3: Electrostatic free energies of proteins 4pti and 2ovo computed with MIBPB solver: $\epsilon_1=4$, 8, 20, $\epsilon_2=80$, ion concentration = 0.15M, MSMS density =10 vertices/Å^2; values at $h=1$, $h=0.5$, and $h=0.25$ show difference from values at $h=0.125$ in kcal/mol; relative error in percentage is benchmarked with results at the finest mesh $h=0.125$.

	$\epsilon_1=4$				$\epsilon_1=8$				$\epsilon_1=20$			
h	1.0	0.5	0.25	0.125	1.0	0.5	0.25	0.125	1.0	0.5	0.25	0.125
4pti	-15.8	+1.9	+0.4	-5887.9	-5.6	+1.3	+0.3	-2933.9	-0.4	+0.8	+0.2	-1162.6
err.(%)	0.27	0.03	0.01	–	0.19	0.05	0.01	–	0.03	0.07	0.02	–
2ovo	-13.9	+1.5	+0.3	-4326.3	-5.3	+1.0	+0.2	-2154.4	-0.7	+0.6	+0.1	-852.4
err.(%)	0.32	0.03	0.01	–	0.25	0.05	0.01	–	0.08	0.06	0.01	–

From Table 3, in each selected dielectric constants ϵ_1, we can see the energies converge to the finest grid $h=0.125$ from left to right with the decrement of h. In the table, in the row containing the PDB ID, we reported the free energy at $h=0.125$ and the difference from that at $h=1.0$,0.5,0.25. In the row below that, we reported the relative error using values at $h=0.125$ as the benchmark. From the table, we can see even when the coarsest mesh $h=1.0$ is used, the relative error using result from $h=0.125$ as the benchmark is still less than 0.5%, which justifies our choice of $h=1.0$ for rapidly solving PB equation for electrostatic free energies and computing pK_a based on these energies.

4.2 pKa computation

We followed the procedure as described in the previous section to compute pK_a values of protein 4pti with 58 residues and 18 titration sites and of protein 2ovo with 56 residues and 15 titration sites. The reason that we choose these relatively smaller proteins is due to the fact we use explicit formulation as opposed to the popular Monto Carlo (MC) simulation for the site-site interactions [1, 12]. The explicit approach has costs increased as 2^{N_t} thus is prohibitively expensive for larger proteins. In this paper, our focus is to demonstrate the accuracy of MIBPB in returning the electrostatic free energy subject to a given protein structure and charge distribution for pK_a calculation. Thus we leave the implementation and enhancement of the MC approach in the future work. In this section, our computed pK_a results are compared with the experimental values from [21] as referred by [1].

Table 4: Computed pK_a values for protein 4pti at various solute dielectric constants: $\epsilon_1 = 4,8,20$, solvent dielectric constant $\epsilon_2 = 80$, ion concentration = 0.15M, MSMS density =10 vertices/Å^2.

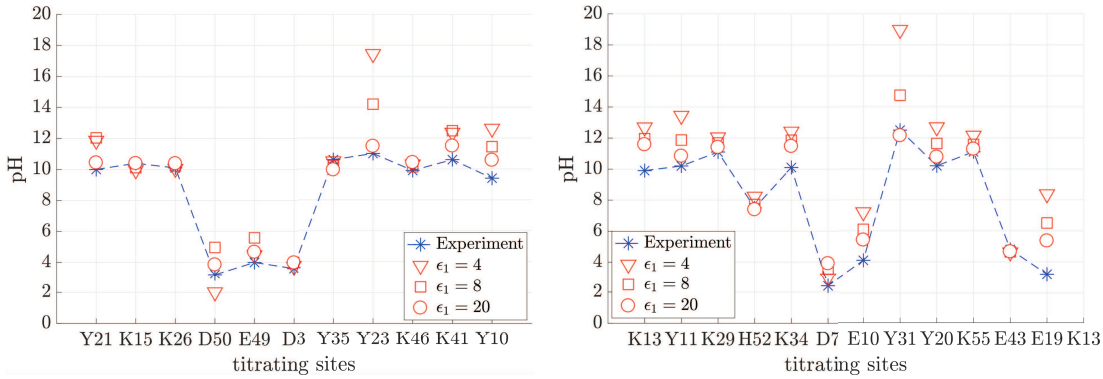
residue	expt. pK_a	intrinsic pK_a			pK_a		
		$\epsilon_1=4$	$\epsilon_1=8$	$\epsilon_1=20$	$\epsilon_1=4$	$\epsilon_1=8$	$\epsilon_1=20$
TYR21	10.0	11.08	10.49	10.13	11.86	12.04	10.43
GLU7	N/A	4.39	4.49	4.57	4.59	4.98	4.60
LYS15	10.4	10.19	10.42	10.59	9.96	10.12	10.39
LYS26	10.1	10.23	10.43	10.56	10.05	10.23	10.38
ASP50	3.2	0.48	2.17	3.21	2.00	4.97	3.85
ARG1	N/A	19.78	16.56	14.63	19.78	16.56	14.63
ARG53	N/A	13.02	12.90	12.86	12.74	12.39	12.70
ARG42	N/A	12.13	12.38	12.54	12.18	12.57	12.31
ARG39	N/A	13.49	12.94	12.63	13.34	12.64	12.50
ARG17	N/A	12.21	12.23	12.25	11.95	11.78	12.12
GLU49	4.0	3.76	4.11	4.32	4.48	5.59	4.67
ASP3	3.6	3.63	3.77	3.92	3.73	3.95	3.98
TYR35	10.6	10.45	10.13	9.92	10.50	10.21	9.95
TYR23	11.0	17.00	13.41	11.27	17.44	14.19	11.47
LYS46	9.9	11.07	11.09	11.08	10.30	10.14	10.42
LYS41	10.6	12.16	11.69	11.43	12.35	12.47	11.49
TYR10	9.4	12.54	11.26	10.49	12.61	11.44	10.57
ARG20	N/A	16.84	15.03	13.87	16.78	14.97	13.82

The results of computed pK_a for the two proteins are included in Tables 4 and 5. The tables first list the experimental pK_a values if they are available in the literature [1, 21], followed by computed intrinsic pK_a and pK_a values at various solute dielectric constants. From the tables, we can see that the computed intrinsic pK_a and pK_a values are essentially close to the experimental results. It seems that including the site-site interaction does not significantly improve the computed pK_a toward the experimental pK_a. In order to reveal the embedded information from the tabular data, we plot the computed pK_a values in Fig. 3 compared with available experimental pK_a values.

Fig. 3(a) and (b) depict computed pK_a result from protein 4pti and 2ovo, respectively. Since patterns in both subplots are similar, we explain them in a uniform manner. In Fig. 3(a-b), the vertical axis is the pK_a values and the horizontal axis is the titration sites. To save space, on x-axis we use one-letter representation of the amino acids, followed by their residue IDs. On these two subplots, the blue stars connected by the dashed line are the results from experiment [1, 21]. The red triangles, square, and circles are computed pK_a using solute dielectric constants $\epsilon = 4,8,16$ respectively. From both subplots, we can see that the computed pK_a are in line with the experimental pK_a and it is quite obvious that using larger solute dielectric constants produced computed pK_a values which are

Table 5: Computed pK_a values for protein 2ovo at various solute dielectric constants: $\epsilon_1 = 4,8,20$, solvent dielectric constant $\epsilon_2 = 80$, ion concentration = 0.15M, MSMS density =10 vertices/Å^2.

residue	expt. pK_a	intrinsic pK_a			pK_a		
		$\epsilon_1=4$	$\epsilon_1=8$	$\epsilon_1=20$	$\epsilon_1=4$	$\epsilon_1=8$	$\epsilon_1=20$
LYS13	9.9	12.59	12.07	11.73	12.69	11.96	11.58
ASP27	N/A	10.74	7.69	5.81	6.79	5.71	5.02
TYR11	10.2	13.40	11.76	10.73	13.44	11.87	10.84
LYS29	11.1	11.75	11.53	11.37	12.05	11.68	11.40
HIS52	7.5	8.03	7.64	7.41	8.23	7.72	7.39
LYS34	10.1	12.79	12.14	11.73	12.42	11.86	11.47
ASP7	2.4	2.00	3.01	3.66	2.95	3.54	3.91
GLU10	4.1	5.19	5.08	4.99	7.21	6.10	5.42
TYR31	$>$12.5	18.97	14.75	12.12	18.97	14.75	12.15
TYR20	10.2	11.01	10.57	10.26	12.69	11.64	10.77
LYS55	11.1	12.15	11.69	11.43	12.15	11.58	11.28
GLU43	4.8	4.66	4.69	4.70	4.62	4.65	4.66
GLU19	3.2	5.73	5.21	4.87	8.37	6.51	5.36
ARG21	N/A	12.54	12.51	12.50	12.88	12.50	12.47



(a) (b)

Figure 3: Computed pK_a at various solute dielectric constants $\epsilon_1 = 4,8,20$ vs experimental pK_a for protein 4pti (a) and protein 2ovo (b).

closer to the experimental results. This conclusion has been advocated by some previous research results [12, 56].

As explained in the previous section, after receiving all the energies from Eq. (3.5), the probability of a titration site to be protonated at a given pH can be computed by using Eq. (3.4). For each titration site, we sampled the probability at pH from 0 to 14 with step size 0.2 and then use cubic spline to interpolate the pH when probability=0.5. By doing

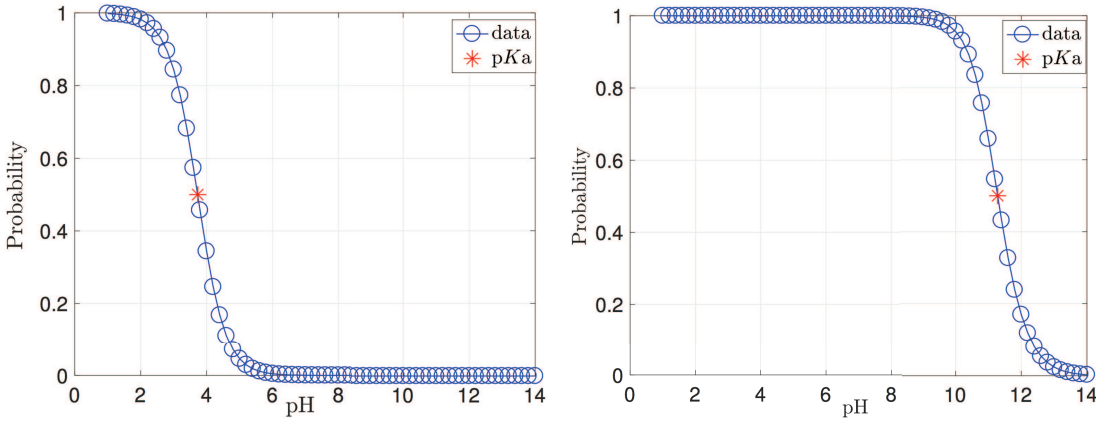


Figure 4: titration curves for (a) titration site ASP3 from protein 4pti ($\epsilon_1=4$): pK_a (computed)=3.73 vs. pK_a (experiment)=3.6 and (b) titration site LYS55 from protein 2ovo ($\epsilon_1=20$): pK_a (computed)=11.28 vs. pK_a (experiment)=11.3.

this, each titration site from every protein at a solute dielectric constant has a titration curve as shown in Fig. 4. Here we choose two examples, one as shown in Fig. 4(a) is from titration site ASP3 from protein 4pti at $\epsilon_1=4$, which produces pK_a = 3.73 against the 3.6 experiment value. The other example as shown in Fig. 4(b) is from titration site LYS55 from protein 2ovo at $\epsilon_1=20$, which produces pK_a = 11.28 against the 11.3 experiment value. As we know Aspartic Acid (ASP) is acidic amino acid while Lysine (LYS) is basic amino acid as two very polar sites whose pK_a computation is very sensitive and our computed values are sufficiently accurate.

5 Conclusion

In this paper, we theoretically computed pK_a values of proteins using the electrostatic free energies. The energies are computed by solving the Poisson-Boltzmann model using our recently developed 2nd order accurate MIBPB solver [26]. The advantage of our MIBPB solver is at its accurate results even at coarse mesh such as $h=1.0$. This is an important advantage since the MIBPB solver will be called for $1+2N_t+\frac{1}{2}(N_t^2-N_t)$ times for full-atom protein structures with variation of the charge distribution. In addition, since the algorithm is implemented under the assumption that protonation or deprotonation does not produce the structural change, charge distribution is the key component in pK_a computation. MIBPB regularizes charge singularity analytically using the Green's function based decomposition thus could more accurately capture sensitive energy changes caused by changes in charge distribution.

In addition to calling the MIBPB solver for calculating electrostatic free energies given structure and charge distribution, we pipelined the entire procedure of pK_a computation

by using Python from downloading PDB structure, assigning charge distribution and protonation states under selected force fields, to computing intrinsic pK_a and then pK_a by including the site-site interactions. To this end, the users input the PDB ID of a target protein and can receive accurate pK_a values on all titration sites quickly. We will make the Python wrapper and the binary code of MIBPB available on author's public website. From the numerical results, our calculation of pK_a is accurate and efficient.

PB model based pK_a values are affected to many different factors such as the choices of dielectric constants and ionic strength, force fields, the initial PDB files, as well as the accuracy of the PB solver. Among these factors, the more disputative ones are the model, the force field, and the protein structure. MIBPB however can reduce the error in solving PB equation to the minimum. We next will move to the topic of using position dependent dielectric constants to enhance pK_a computation based on established schemes.

Acknowledgments

The authors would like to thank Dr. Robert Krasny for useful discussions. This work is founded by National Science Foundation under the contracts DMS-1318898 and DMS-1418957. The computing facility is provided by SMU center of supercomputing. Undergraduate research assistant Jingzhen Hu is partially supported by SMU Hamilton Scholarship and Summer Research Assistantship.

References

[1] K. L. Ho, *Fast direc methods for molecular electrostatics*. PhD thesis, New York University, 2012.
[2] M. F. Jeng and H. J. Dyson, "Direct measurement of the aspartic acid 26 pka for reduced escherichia coli thioredoxin by 13c nmr.," *Biochemistry*, vol. 35, pp. 1–6, Jan 1996.
[3] K. Bartik, C. Redfield, and C. M. Dobson, "Measurement of the individual pka values of acidic residues of hen and turkey lysozymes by two-dimensional 1h nmr.," *Biophysical Journal*, vol. 66, pp. 1180–1184, 04 1994.
[4] E. Fermi, *Thermodynamics*. Dover Publications, Inc. New York, 1936.
[5] M. Gunner and N. Baker, "Chapter one - continuum electrostatics approaches to calculating pkas and ems in proteins," in *Computational Approaches for Studying Enzyme Mechanism Part B* (G. A. Voth, ed.), vol. 578 of *Methods in Enzymology*, pp. 1 – 20, Academic Press, 2016.
[6] E. Alexov, E. L. Mehler, N. Baker, A. M. Baptista, Y. Huang, F. Milletti, J. E. Nielsen, D. Farrell, T. Carstensen, M. H. M. Olsson, J. K. Shen, J. Warwicker, S. Williams, and J. M. Word, "Progress in the prediction of pka values in proteins.," *Proteins*, vol. 79, pp. 3260–3275, Dec 2011.
[7] L. Wang, L. Li, and E. Alexov, "pka predictions for proteins, rnas, and dnas with the gaussian dielectric function using delphi pka," *Proteins: Structure, Function, and Bioinformatics*, vol. 83, no. 12, pp. 2186–2197, 2015.
[8] J. Antosiewicz, J. A. McCammon, and M. K. Gilson, "The determinants of pK_as in proteins," *Biochemistry*, vol. 35, no. 24, pp. 7819–7833, 1996.
[9] D. Bashford and M. Karplus, "pK_a's of ionizable groups in proteins: atomic detail from a continuum electrostatic model," *Biochemistry*, vol. 29, no. 44, pp. 10219–25, 1990.

[10] F. Fogolari, A. Brigo, and H. Molinari, "The Poisson-Boltzmann equation for biomolecular electrostatics: a tool for structural biology," *Journal of Molecular Recognition*, vol. 15, no. 6, pp. 377–92, 2002.
[11] R. E. Georgescu, E. G. Alexov, and M. R. Gunner, "Combining conformational flexibility and continuum electrostatics for calculating pkas in proteins," *Biophysical Journal*, vol. 83, no. 4, pp. 1731 – 1748, 2002.
[12] A. H. Juffer, P. Argos, and H. J. Vogel, "Calculating acid-dissociation constants of proteins using the boundary element method," *The Journal of Physical Chemistry B*, vol. 101, no. 38, pp. 7664–7673, 1997.
[13] J. E. Nielsen and J. A. McCammon, "On the evaluation and optimization of protein x-ray structures for pka calculations," *Protein Science*, vol. 12, no. 2, pp. 313–326, 2003.
[14] J. E. Nielsen and G. Vriend, "Optimizing the hydrogen-bond network in Poisson-Boltzmann equation-based pK(a) calculations," *Proteins*, vol. 43, no. 4, pp. 403–412, 2001.
[15] J. Warwicker, "Improved pka calculations through flexibility based sampling of a water-dominated interaction scheme," *Protein Science*, vol. 13, no. 10, pp. 2793–2805, 2004.
[16] A. S. Yang, M. R. Gunner, R. Sampogna, K. Sharp, and B. Honig, "On the calculation of pK(a)s in proteins," *Proteins-Structure Function and Genetics*, vol. 15, no. 3, pp. 252–265, 1993.
[17] T. Meyer and E.-W. Knapp, "pka values in proteins determined by electrostatics applied to molecular dynamics trajectories," *Journal of Chemical Theory and Computation*, vol. 11, no. 6, pp. 2827–2840, 2015. PMID: 26575575.
[18] K. P. Kilambi and J. J. Gray, "Rapid calculation of protein pka values using rosetta.," *Biophys J*, vol. 103, pp. 587–595, Aug 2012.
[19] D. Riccardi, P. Schaefer, and Q. Cui, "pka calculations in solution and proteins with QM/MM free energy perturbation simulations: a quantitative test of QM/MM protocols," *The Journal of Physical Chemistry B*, vol. 109, no. 37, pp. 17715–17733, 2005.
[20] L. J. Gosink, E. A. Hogan, T. C. Pulsipher, and N. A. Baker, "Bayesian model aggregation for ensemble-based estimates of protein pka values," *Proteins: Structure, Function, and Bioinformatics*, vol. 82, no. 3, pp. 354–363, 2014.
[21] H. Li, A. D. Robertson, and J. H. Jensen, "Very fast empirical prediction and rationalization of protein pka values," *Proteins: Structure, Function, and Bioinformatics*, vol. 61, no. 4, pp. 704–721, 2005.
[22] 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," *J. Comput. Phys.*, vol. 213, no. 1, pp. 1–30, 2006.
[23] S. Yu, W. Geng, and G. W. Wei, "Treatment of geometric singularities in implicit solvent models," *Journal of Chemical Physics*, vol. 126, p. 244108, 2007.
[24] W. Geng, S. Yu, and G. W. Wei, "Treatment of charge singularities in implicit solvent models," *Journal of Chemical Physics*, vol. 127, p. 114106, 2007.
[25] D. Chen, Z. Chen, C. Chen, W. H. Geng, and G. W. Wei, "MIBPB: A software package for electrostatic analysis," *J. Comput. Chem.*, vol. 32, pp. 657 – 670, 2011.
[26] W. Geng and S. Zhao, "A two-component matched interface and boundary (mib) regularization for charge singularity in implicit solvation," *J. Comput. Phys.*, accepted, 2017.
[27] W. Geng, "A boundary integral Poisson–Boltzmann solvers package for solvated bimolecular simulations," *Molecular Based Mathematical Biology*, vol. 3, pp. 43–58, 2015.
[28] M. J. Holst, *The Poisson-Boltzmann Equation: Analysis and Multilevel Numerical Solution*. PhD thesis, UIUC, 1994.
[29] B. Honig and A. Nicholls, "Classical electrostatics in biology and chemistry," *Science*,

vol. 268, no. 5214, pp. 1144–9, 1995.
[30] M. E. Davis, J. D. Madura, J. Sines, B. A. Luty, S. A. Allison, and J. A. McCammon, "Diffusion-controlled enzymatic reactions," *Methods in Enzymology*, vol. 202, pp. 473–497, 1991.
[31] W. Im, D. Beglov, and B. Roux, "Continuum solvation model: electrostatic forces from numerical solutions to the Poisson-Boltzmann equation," *Computer Physics Communications*, vol. 111, no. 1-3, pp. 59–75, 1998.
[32] N. A. Baker, D. Sept, S. Joseph, M. J. Holst, and J. A. McCammon, "Electrostatics of nanosystems: Application to microtubules and the ribosome," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 98, no. 18, pp. 10037–10041, 2001.
[33] R. Luo, L. David, and M. K. Gilson, "Accelerated Poisson-Boltzmann calculations for static and dynamic systems," *Journal of Computational Chemistry*, vol. 23, no. 13, pp. 1244–53, 2002.
[34] Q. Cai, J. Wang, H.-K. Zhao, and R. Luo, "On removal of charge singularity in poissonboltzmann equation," *The Journal of Chemical Physics*, vol. 130, no. 14, 2009.
[35] W. Deng, X. Zhufu, J. Xu, and S. Zhao, "A new discontinuous galerkin method for the nonlinear poisson-boltzmann equation," *Applied Mathematics Letters*, vol. 257, pp. 1000–1021, 2015.
[36] W. Geng and R. Krasny, "A treecode-accelerated boundary integral Poisson-Boltzmann solver for electrostatics of solvated biomolecules," *Journal of Computational Physics*, vol. 247, no. 0, pp. 62 – 78, 2013.
[37] A. Juffer, B. E., B. van Keulen, A. van der Ploeg, and H. Berendsen, "The electric potential of a macromolecule in a solvent: a fundamental approach," *J. Comput. Phys.*, vol. 97, pp. 144–171, 1991.
[38] J. Liang and S. Subranmaniam, "Computation of molecular electrostatics with boundary element methods," *Biophys. J.*, vol. 73, pp. 1830–1841, 1997.
[39] A. H. Boschitsch, M. O. Fenley, and H.-X. Zhou, "Fast boundary element method for the linear Poisson-Boltzmann equation," *The Journal of Physical Chemistry B*, vol. 106, no. 10, pp. 2741–2754, 2002.
[40] B. Lu, X. Cheng, and J. A. McCammon, "A new-version-fast-multipole-method accelerated electrostatic calculations in biomolecular systems," *Journal of Computational Physics*, vol. 226, no. 2, pp. 1348 – 1366, 2007.
[41] L. Greengard, D. Gueyffier, P.-G. Martinsson, and V. Rokhlin, "Fast direct solvers for integral equations in complex three-dimensional domains," *Acta Numerica*, vol. 18, pp. 243–275, 005 2009.
[42] C. Bajaj, S.-C. Chen, and A. Rand, "An efficient higher-order fast multipole boundary element solution for poissonboltzmann-based molecular electrostatics," *SIAM Journal on Scientific Computing*, vol. 33, no. 2, pp. 826–848, 2011.
[43] B. Zhang, B. Lu, X. Cheng, J. Huang, N. P. Pitsianis, X. Sun, and J. A. McCammon, "Mathematical and numerical aspects of the adaptive fast multipole poisson-boltzmann solver," *Communications in Computational Physics*, vol. 13, pp. 107–128, 001 2013.
[44] B. Z. Lu, Y. C. Zhou, M. J. Holst, and J. A. McCammon, "Recent progress in numerical methods for the Poisson-Boltzmann equation in biophysical applications," *Communications in Computational Physics*, vol. 3, no. 5, pp. 973–1009, 2008.
[45] N. A. Baker, "Improving implicit solvent simulations: a Poisson-centric view," *Current Opinion in Structural Biology*, vol. 15, no. 2, pp. 137–43, 2005.
[46] S. Zhao and G. W. Wei, "High-order FDTD methods via derivative matching for Maxwell's equations with material interfaces," *J. Comput. Phys.*, vol. 200, no. 1, pp. 60–103, 2004.

[47] Y. C. Zhou and G. W. Wei, "On the fictitious-domain and interpolation formulations of the matched interface and boundary (MIB) method," *J. Comput. Phys.*, vol. 219, no. 1, pp. 228–246, 2006.
[48] K. A. Sharp and B. Honig, "Calculating total electrostatic energies with the nonlinear Poisson-Boltzmann equation," *Journal of Physical Chemistry*, vol. 94, pp. 7684–7692, 1990.
[49] M. K. Gilson, M. E. Davis, B. A. Luty, and J. A. McCammon, "Computation of electrostatic forces on solvated molecules using the Poisson-Boltzmann equation," *Journal of Physical Chemistry*, vol. 97, no. 14, pp. 3591–3600, 1993.
[50] W. Geng and G. W. Wei, "Multiscale molecular dynamics using the matched interface and boundary method," *J Comput. Phys.*, vol. 230, no. 2, pp. 435–457, 2011.
[51] T. J. Dolinsky, P. Czodrowski, H. Li, J. E. Nielsen, J. H. Jensen, G. Klebe, and N. A. Baker, "Pdb2pqr: expanding and upgrading automated preparation of biomolecular structures for molecular simulations," *Nucleic Acids Research*, vol. 35, 2007.
[52] Y. Nozaki and C. Tanford, "Examination of titration behavior," *Methods in Enzymology*, vol. 11, pp. 715 – 734, 1967.
[53] M. F. Sanner, A. J. Olson, and J. C. Spehner, "Reduced surface: An efficient way to compute molecular surfaces," *Biopolymers*, vol. 38, pp. 305–320, 1996.
[54] M. Marquart, J. Walter, J. Deisenhofer, W. Bode, and R. Huber, "The geometry of the reactive site and of the peptide groups in trypsin, trypsinogen and its complexes with inhibitors," *Acta Crystallographica Section B*, vol. 39, pp. 480–490, Aug 1983.
[55] W. Bode, O. Epp, R. Huber, M. Laskowski, and W. Ardelt, "The crystal and molecular structure of the third domain of silver pheasant ovomucoid (omsvp3)," *European Journal of Biochemistry*, vol. 147, no. 2, pp. 387–395, 1985.
[56] J. Antosiewicz, J. McCammon, and M. K. Gilson, "Prediction of ph-dependent properties of proteins," *Journal of Molecular Biology*, vol. 238, no. 3, pp. 415 – 436, 1994.