Journal of Computational Physics 275 (2014) 294–309

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

New solution decomposition and minimization schemes for Poisson–Boltzmann equation in calculation of biomolecular electrostatics

Dexuan Xie

Department of Mathematical Sciences, University of Wisconsin–Milwaukee, Milwaukee, WI 53211, USA

ARTICLE INFO

Article history:
Received 11 September 2013
Received in revised form 3 May 2014
Accepted 3 July 2014
Available online 16 July 2014

Keywords:
Poisson–Boltzmann equation
Finite element method
Variational minimization
Biomolecular electrostatics
FEniCS project

ABSTRACT

The Poisson–Boltzmann equation (PBE) is one widely-used implicit solvent continuum model in the calculation of electrostatic potential energy for biomolecules in ionic solvent, but its numerical solution remains a challenge due to its strong singularity and nonlinearity caused by its singular distribution source terms and exponential nonlinear terms. To effectively deal with such a challenge, in this paper, new solution decomposition and minimization schemes are proposed, together with a new PBE analysis on solution existence and uniqueness. Moreover, a PBE finite element program package is developed in Python based on the `FEniCS` program library and `GAMer`, a molecular surface and volumetric mesh generation program package. Numerical tests on proteins and a nonlinear Born ball model with an analytical solution validate the new solution decomposition and minimization schemes, and demonstrate the effectiveness and efficiency of the new PBE finite element program package.

1. Introduction

The Poisson–Boltzmann equation (PBE) is one widely-used implicit solvent continuum model in the calculation of electrostatic potential energy for biomolecule in ionic solvent [1–4]. However, its numerical solution is very challenging due to strong singularity and nonlinearity caused by its singular distribution and exponential nonlinear terms. In the past two decades, these challenges were addressed via typical numerical techniques (such as finite difference, finite element, and boundary element methods) and popular linear and nonlinear iterative methods (such as the successive over-relaxation method, the conjugate gradient method, the inexact-Newton method, the multigrid method, and numerical optimization methods) [4–14]. Several PBE program packages and web-based resources were developed, which include DelPhi [6,15], MEAD [16], APBS [17,18], PBE solver modules in the biomolecular modeling and simulation programs AMBER [19,20], CHARMM [21–23], and NAMD [17,24], making the PBE model a powerful simulation tool in the study of biomolecular structure, biological function, catalytic activity, ligand association, and rational drug design [1,25–27].

To further improve current PBE mathematical analysis, in this paper, we first present a novel PBE solution decomposition to split the PBE solution u into three parts within both the solute domain D_p and solvent domain D_s. These three parts, G, Ψ, and $\tilde{\Phi}$ (see Theorem 3.1), correspond to electrostatic contributions from the biomolecular charges, the boundary and interface conditions, and the ionic solvent charges, respectively. Here, G is a known function collecting all the singular points of u while both Ψ and $\tilde{\Phi}$ are twice differentiable in D_p and D_s. Hence, u can be found through calculating Ψ

E-mail address: dxie@uwm.edu.

http://dx.doi.org/10.1016/j.jcp.2014.07.012
0021-9991/

and $\tilde{\Phi}$ without involving any singular difficulty. Note that our solution decomposition differs from those in [28–31]. In the decomposition from [28], u was split within D_p only. In the decompositions from [29–31], Ψ and $\tilde{\Phi}$ were defined by elliptic boundary value problems with discontinuous coefficients, which had definitions only in the weak sense. Our Ψ and $\tilde{\Phi}$ are defined by elliptic interface problems, which are well defined in both strong and weak senses. All of the previous solution decompositions were performed only for a symmetric 1:1 ionic solution. In contrast, our solution decomposition works for a solvent containing any number of ionic species.

As an application of this PBE solution decomposition, we then construct new finite element solution decomposition and minimization schemes for solving PBE without any singular and "blow-up" difficulty. To do so, we begin with a review of the PBE model for a biomolecule (protein or nucleic acids) immersed in an ionic solvent containing n ionic species. In this concise review, the PBE model is clearly described in both SI (Systéke International) units and electrostatic units. Here, the values and units of all involved physical parameters are given for the convenience of study. We then show that the PBE model using either electrostatic or SI units can be transformed into the same dimensionless form (see (5)) when the length is measured in angstroms (Å). Hence, we only need to consider this dimensionless PBE model for the calculations of biomolecular electrostatics.

Our PBE solution decomposition and proof on PBE solution existence and uniqueness are presented for the dimensionless PBE model (see Theorems 3.1, 4.1, and 4.2). Note that a Lagrange finite element space can be a finite dimensional subspace of a Sobolev function space. Hence, a PBE finite element solution decomposition scheme (see Algorithm 1) can be directly followed from this novel PBE solution decomposition. As a finite element method, it includes the interface conditions of the PBE model naturally so that it can produce a numerical PBE solution with a higher numerical accuracy than a finite difference method.

Typically, a nonlinear boundary value problem is solved numerically as a system of nonlinear algebraic equations. To achieve a global convergence, an "artificial" merit function (see (39)) is often employed to yield a trust-region, line-search, or inexact Newton method [10,32–34]. Because of our PBE solution decomposition, $\tilde{\Phi}$ is found to be a unique solution of a variational minimization problem with a target functional J over a Sobolev function space, and the first and second derivatives of J are available (see Theorem 4.2). Hence, J is a "natural" merit function for us to use to develop an efficient Newton-type minimization algorithm for solving a nonlinear boundary value problem of $\tilde{\Phi}$. In our early work [14], we showed one minimization protocol for solving a system of PBE mortar finite element equations to be much more efficient and effective than a popular nonlinear iterative solver — a subspace trust region Newton method [32,33]. In this paper, we intend to extend this work to the case of a Lagrange finite element approximation to $\tilde{\Phi}$.

With our PBE solution decomposition, we propose a simple treatment to deal with a potential "blow-up" problem caused by PBE exponential nonlinear terms without affecting the accuracy of a PBE numerical solution. So far, we did not see any paper that addressed such a "blow-up" issue. We only encountered one treatment on the "blow-up" issue in a code survey of the program package APBS. In our simple treatment, we first construct a modified Newton bilinear form (see (40)) using a function truncation strategy. A new modified Newton minimization scheme is then developed through solving this modified Newton bilinear form by the preconditioned conjugate gradient (PCG) method with incomplete LU (ILU) preconditioning. Our function truncation strategy may not affect any accuracy of a PBE finite element solution since a possible modification to the target functional J or its derivatives happens only in the early stage of a minimum search process. To reflect a possible affect of a modified J to the new modified Newton minimization scheme, a special iteration test (see (41)) is added to make the modified Newton minimization scheme more robust. Eventually, our new modified Newton minimization scheme becomes a descent search method so that its convergence can be followed directly from the descent search minimization theory [34,35].

A combination of the PBE solution decomposition scheme with the modified Newton minimization scheme leads to a new effective PBE finite element solver. In this paper, we program it in Python as a new PBE finite element program package for a protein in a symmetric 1:1 ionic solvent based on the FEniCS finite element library [36] and a molecular surface and volumetric mesh generation program package, `GAMer` [37]. As a Python program package, our new PBE program package is easy to be used and portable on different computer operating systems. Due to the FEniCS finite element library, various finite element methods and various direct and iterative linear solvers become available for calculating Ψ and $\tilde{\Phi}$ numerically. We adapted `GAMer` as a Python module so that a tetrahedral mesh can be generated within our PBE program package to match the need of a FEniCS finite element solver. In addition, to speed up calculation, we wrote Fortran subroutines for computing the mesh node values of G, ∇G, and our modified hyperbolic functions (see (49)), and converted them as Python modules. All the related parameters from the PBE model, `DOLFIN`, and `GAMer` are collected into one parameter file, with which we can easily control solution accuracy and mesh quality. In this way, a protein file is the only input file for an implementation of our new PBE finite element program package.

With this new PBE finite element Python program package, we first made numerical tests on a nonlinear Born ball model with analytical solution using linear, quadratic, and cubic finite element methods. Numerical results validated the PBE solution decomposition scheme and our new PBE finite element program package. We then conducted numerical experiments on a protein suite with the number of atoms up to 6062 in a linear finite element method. Numerical results demonstrated the effectiveness and efficiency of the modified Newton minimization scheme and the high performance of our new PBE finite element program package. For example, in a test of protein represented in the PDB file 4PTI, the total computer CPU time was only about 31 seconds on one 2.3 GHz Intel Core i7 of a MacBook Pro, which included the time spent on the generation of a finite element mesh with 33 572 vertices and 191 372 tetrahedra.