Journal of Computational Physics 298 (2015) 636–651

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

A new finite element and finite difference hybrid method for computing electrostatics of ionic solvated biomolecule

Jinyong Ying, Dexuan Xie *

Department of Mathematical Sciences, University of Wisconsin-Milwaukee, Milwaukee, WI, 53201, USA

ARTICLE INFO

Article history:
Received 24 November 2014
Received in revised form 10 June 2015
Accepted 24 June 2015
Available online 29 June 2015

Keywords:
Poisson–Boltzmann equation
Finite element method
Finite difference method
Biomolecular electrostatics
Domain decomposition
Multigrid preconditioning
Solvation free energy

ABSTRACT

The Poisson–Boltzmann equation (PBE) is one widely-used implicit solvent continuum model for calculating electrostatics of ionic solvated biomolecule. In this paper, a new finite element and finite difference hybrid method is presented to solve PBE efficiently based on a special seven-overlapped box partition with one central box containing the solute region and surrounded by six neighboring boxes. In particular, an efficient finite element solver is applied to the central box while a fast preconditioned conjugate gradient method using a multigrid V-cycle preconditioning is constructed for solving a system of finite difference equations defined on a uniform mesh of each neighboring box. Moreover, the PBE domain, the box partition, and an interface fitted tetrahedral mesh of the central box can be generated adaptively for a given PQR file of a biomolecule. This new hybrid PBE solver is programmed in `C`, `Fortran`, and `Python` as a software tool for predicting electrostatics of a biomolecule in a symmetric 1:1 ionic solvent. Numerical results on two test models with analytical solutions and 12 proteins validate this new software tool, and demonstrate its high performance in terms of CPU time and memory usage.

1. Introduction

Electrostatic interactions are important in understanding the structure and biological function of biomolecules, catalytic activity, ligand association, and pK_a values [1–4]. The Poisson–Boltzmann equation (PBE) is one widely-used implicit solvent continuum model for calculating electrostatics of biomolecules (protein or nucleic acids) in ionic solvent [5–7]. A lot of work has been done in the development of PBE numerical solvers and program packages based on finite difference, finite element, and boundary integral equation approaches [8–13]. The popular PBE program packages and web-based resources, such as APBS [10], DelPhi [14], PBEQ [12,15], and UHBD [16], have become powerful simulation aides in the study of biomolecular structure and function, biomolecule-ligand association, ion channel modeling, and computer-aided drug design.

To improve the accuracy of PBE numerical solutions, we recently developed a solution decomposition PBE solver (SDPB) using finite element and minimization techniques [13]. Instead of solving PBE directly, in SDPB, two other interface problems — one linear interface problem for Ψ and one nonlinear interface problem for $\tilde{\Phi}$ — are solved to yield a numerical solution u of PBE as a sum of a given function G with Ψ and $\tilde{\Phi}$. Since G collects all the singularity points of u, both Ψ and $\tilde{\Phi}$ become twice continuously differentiable within both the solute and solvent regions. Thus, their numerical approximations can be calculated much more easily and less expensively than directly calculating u, which is singular at each atomic position $\mathbf{r}_j$, to reach a required numerical accuracy. We note that in DelPhi, APBS and PBEQ, a finite difference method

* Corresponding author.
E-mail address: dxie@uwm.edu (D. Xie).

http://dx.doi.org/10.1016/j.jcp.2015.06.016
0021-9991/

defined on a uniform mesh accelerated by geometric multigrid techniques is the major algorithm for solving PBE directly. Its mesh size h may become very small to catch the solution behavior of PBE around each singularity point $\mathbf{r}_j$, making the finite difference system too large to be solved due to the memory limitation of computers. Hence, it is critical to solve PBE indirectly through computing Ψ and $\tilde{\Phi}$.

Another feature of SDPB is to use an unstructured interface fitted tetrahedral mesh to approximate the interface Γ between the solute and solvent regions. Such a treatment can yield a numerical solution of PBE in high accuracy in comparison to the case of a uniform finite difference mesh [17]. However, using an unstructured mesh may cause efficient geometric multigrid techniques no longer to work, and require extra memory locations to store the mesh data and the nonzero entries of the coefficient matrix of each involved linear system.

In order to confine the disadvantages while retaining the advantages of SDPB, in this paper, we present a new finite element and finite difference hybrid method based on the Schwartz domain decomposition approach [18,19]. This new hybrid PBE solver was motivated from the fact that the linear variational problem for determining the search direction p_k of the modified Newton minimization algorithm from SDPB can be reformulated as a linear elliptic interface boundary value problem (see Theorem 3.1). From this fact it implies that the modified Newton minimization algorithm can also be implemented based on the finite difference approach. The only change to be made is to solve the linear interface problems of Ψ (see (5)) and p_k (see (11)) by a finite difference method. To combine the advantages of finite element and finite difference approaches together, we propose an overlapped box iterative method for solving the linear interface problems of Ψ and p_k based on a special partition of the whole domain Ω into seven overlapped boxes, in which one central box contains the solute region D_p and is surrounded by the six neighboring boxes (see Fig. 1 for an illustration). Since each neighboring box is a part of the solvent region D_s only, the corresponding equations of Ψ and p_k are reduced to the regular linear elliptic boundary value problems. Thus, they can be solved by a finite difference method defined on a uniform mesh to enable the usage of efficient geometric multigrid techniques. To do so, we construct the preconditioned conjugate gradient (PCG) method using a multigrid V-cycle preconditioning, called PCG-MG, on each neighboring box while adopting the finite element solver from [13] to solve the interface variational problem on the central box. Using this special overlapped box iterative method, we modify SDPB as a finite element and finite difference hybrid method for solving PBE, which is expected to improve the performance of SDPB significantly in terms of CPU time and memory usage.

For a general purpose, in the paper, we describe the overlapped box iterative method for solving a general interface boundary value problem (see (12)), which contains the interface problems of Ψ and p_k as two particular cases. Here the number of boxes has been set as the smallest number of seven to make the overlapped box iterative method to have a fixed rate of convergence for a fixed value of relaxation parameters ω, since the convergence rate of an overlapped box iterative method may decay with the increment of the number of boxes. The central box has been ordered as the 7th box to employ all the most recent iterates from the six neighboring boxes to update the boundary value function of the interface problem on the central box. Moreover, we propose a scheme for this new hybrid PBE solver to adaptively generate Ω, the seven boxes of Ω, an interface fitted tetrahedral mesh of the central box, and a uniform mesh of each neighboring box according to a given PQR file of a biomolecule to be calculated. In this scheme, the same uniform mesh has been applied to the overlapped part of each box so that the data exchange between any two neighboring boxes can be done easily and efficiently.

We completed the program of our hybrid PBE solver in `C`, `Fortran`, and `Python`. The scheme for the generation of Ω, seven boxes, and meshes was programmed in `C`, along with three new parameters for a user to adjust the size of Ω, the size of the central box, and the mesh size h of the uniform mesh. We modified the molecular surface and volumetric mesh generation program package `GAMer` [20] to make it work for a rectangular domain. We then applied it to our `C` program for the generation of an interface fitted mesh of the central box (see Fig. 2). We programmed the PCG-MG method in `Fortran` without storing any mesh data or coefficient matrices of finite difference systems. In the multigrid V-cycle preconditioning, one forward Gauss–Seidel iteration and one backward Gauss–Seidel iteration were used to define the pre-smoother and post-smoother, respectively. The coarsest grid equation was solved simply by the successive over-relaxation (SOR) method. The PCG using the incomplete LU preconditioning (PCG-ILU) from the `PETSc` library [21] was used to solve each system of finite element equations on the central box. Furthermore, to speedup calculations, we wrote `Fortran` subroutines for calculating the values of G at the mesh nodes of Ω, and the values of ∇G at the mesh nodes of the central box. All of the `Fortran` subroutines and `C` programs were converted to `Python` modules by using the Fortran-to-Python interface generator `f2py` (*http://cens.ioc.ee/projects/f2py2e/*) and `SWIG` (*http://www.swig.org*), respectively. By using those modules, the hybrid PBE solver was programmed in `Python` as a part of the software package SDPB.

To validate our new hybrid PBE solver and program, we made numerical experiments on two test problems with analytical solutions: One is an interface test model problem proposed in [17], and the other one is the nonlinear Born Ball model problem used in [13]. Three nested meshes were constructed for these experiments. Numerical results showed that almost three fourths of the absolute error norms between the numerical and analytical solutions were reduced when the mesh size was reduced by half, which well validated our new hybrid method and program package.

We further demonstrated the performance of our new hybrid PBE solver for 12 proteins (the number of atoms up to 11,439) in terms of CPU time and memory usage. Numerical results showed that our new hybrid PBE solver took the same number of modified Newton iterations to satisfy the iteration termination rule as SDPB did. This confirms that the new hybrid PBE solver retains the convergence rate of SDPB. Due to the efficiency of our special overlapped box iterative method, the new hybrid PBE solver was found to improve the performance of SDPB significantly. In these numerical tests, the total CPU time of SDPB was reduced by 55% to 73%. For example, for a protein with PDB ID 2LZX on a mesh with