PROTEINS: Structure, Function, and Genetics 18:231–245 (1994)

Treatment of Electrostatic Effects in Proteins: Multigrid-Based Newton Iterative Method for Solution of the Full Nonlinear Poisson–Boltzmann Equation

Michael Holst,[1] Richard E. Kozack,[2] Faisal Saied,[1] and Shankar Subramaniam[2,3]
[1]Numerical Computing Group, Department of Computer Science and [2]Center for Biophysics and Computational Biology, Department of Physiology and Biophysics, National Center for Supercomputing Applications, [3]Beckman Institute for Advanced Science and Technology, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801

ABSTRACT **The nonlinear Poisson–Boltzmann equation (NPBE) provides a continuum description of the electrostatic field in an ionic medium around a macromolecule. Here, a novel approach to the solution of the full NPBE is developed. This robust and efficient algorithm combines multilevel techniques with a damped inexact Newton's method. The CPU time required for solution of the full NPBE, which is less than that for standard single-grid approaches in solving the corresponding linearized equation, is proportional to the number of unknowns enabling applications to very large macromolecular systems. Convergence of the method is demonstrated for a variety of protein systems. Comparison of the solutions to the linearized Poisson–Boltzmann equation shows that the damping of the electrostatic field around the charge is increased and that the potential scales logarithmically with charge. The inclusion of the full nonlinearity thus reduces the impact of highly charged residues on protein surfaces and provides a more realistic representation of electrostatic effects. This is demonstrated through calculation of potential around the active site regions of the 1,266-residue tryptophan synthase dimer and in the computation of rate constants from Brownian dynamics calculations in the superoxide dismutase–superoxide and antibody–antigen systems.** © 1994 Wiley-Liss, Inc.

Key words: nonlinear elliptic equations, nonlinear multigrid, inexact Newton methods, damped Newton methods, crambin, BPTI, HyHEL-5, superoxide dismutase, rhinovirus, tryptophan synthase, electrostatic steering, Brownian dynamics, antibody–antigen complex

INTRODUCTION

It is now clear that continuum models of molecules in ionic solutions, which were first proposed in 1923 by Debye and Hückel,[1] are invaluable for studying electrostatic interactions. Since the electrostatic behavior determines to a large degree the structure and kinetics of complex molecules such as proteins, modeling these interactions accurately is an important goal in biophysics[2–5] and molecular dynamics simulations.[6–8] The Debye–Hückel model is depicted in Figure 1. There are three relevant regions in the problem. In the solvent, which is assigned the dielectric constant of water, the charge density of the mobile ions is assumed to be given by a Boltzmann distribution. The molecule, from which ions are excluded, is taken to be a low dielectric region in which point charges are embedded. Often a third region, the Stern layer, is added to simulate the finite size of the ions in the electrolyte. Ions are excluded from the Stern layer, but the dielectric constant is equal to that of the solvent. These assumptions lead to the *nonlinear Poisson–Boltzmann equation* (NPBE), a three-dimensional second order nonlinear elliptic partial differential equation describing the electrostatic potential $\Phi(r)$ at a field position r. If we define a dimensionless potential by $u(r) = e_c\Phi(r)/k_BT$, then the NPBE for a 1:1 electrolyte can be written in the form:

$$-\nabla \cdot [\epsilon(r)\nabla u(r)] + \epsilon(r)\kappa^2(r)\sinh[u(r)] = \frac{e_c^2}{k_BT}\sum_{i=1}^{N_m} z_i\delta(r - r_i) \qquad \text{in } \Omega \subset \mathbb{R}^3, \tag{1}$$

$$u(r) = g(r) \text{ on the boundary } \partial\Omega. \tag{2}$$

The constants e_c, k_B, and T in (1) represent the elementary unit of charge, Boltzmann's constant, and the absolute temperature, respectively. The piecewise constant dielectric is $\epsilon(r)$ and the inverse Debye–Hückel length $\kappa^2(r)$ is given by $\kappa^2 = 2Ie_c^2/\epsilon kT$, where I is the solvent ionic strength. The charge distribution of the molecule is described by the N_m point charges $q_i = z_ie_c$ at positions r_i, yielding the

Received August 16, 1993; revision accepted October 8, 1993.
Address reprint requests to Shankar Subramaniam, Beckman Institute, University of Illinois at Urbana-Champaign, 405 N. Mathews Avenue, Urbana, Illinois 61801.

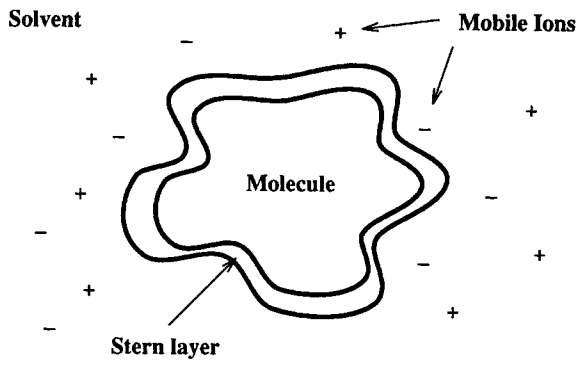

Fig. 1. The Debye–Hückel model for macromolecules in ionic solution.

delta functions in (1). The domain Ω is a rectangular box that encloses the protein and the function $g(r)$ on $\partial\Omega$, the surface of the box, is chosen to reflect the asymptotic behavior of the potential at large distances from the molecule. In practical investigations involving protein electrostatics, it is frequently assumed that u is small compared to unity and therefore the $\sinh(u)$ term can be replaced by u, the first term in its power series expansion. This simplification results in the widely used *linearized Poisson–Boltzmann equation* (LPBE).

Although it can be shown using standard techniques that both the NPBE and LPBE always have unique solutions,[9] closed form expressions for such solutions are quite complex, even in the few simple situations for which they exist.[10] A very early analytical approximation approach to the NPBE for a spherical molecule appears in Brenner and Roberts,[11] whereas analytical approaches to closely related equations can be found in Fujita[12] and Ting et al.[13] Due to advances in computational algorithms and hardware in recent years, several investigations into the efficiency and accuracy of numerical methods for the linearized Poisson–Boltzmann equation (LPBE) have appeared.[14–18] In such approaches, the LPBE is recast as a finite-difference equation on a cubic grid,[19] and iterative matrix techniques are employed to handle the resulting system of algebraic equations. However, difficulties are encountered when standard linear methods are generalized to nonlinear problems, because they are not globally convergent,[20] meaning they are not guaranteed to arrive at the true solution from an arbitrary initial guess. Hence, the NPBE is only now beginning to find use as a tool for studying electrostatic properties, and various numerical methods are being proposed and investigated.[17,21–26] Until now, though, there have been no detailed studies or comparisons of the efficiency and robustness of the proposed numerical methods.

In spite of the increased complexity of the NPBE, there are cases where the linear approximation is inadequate, and the full equation must be solved. As stated above, the LPBE assumes that u is much less than one. However, if we look at the analytic solution at the surface of a spherical molecule, we find that $u = 7.1Q/a(1 + \kappa a)$ at room temperature, where Q is the molecular charge in units of e_c, and a is the radius in Å. It is clear that u can be quite large for reasonable values of Q and a, particularly at lower ionic strengths. The solutions of the NPBE and LPBE also have entirely different scaling properties with respect to molecular charge. By investigating simple systems, we have found that the magnitude of u grows only logarithmically with the magnitude of the charge in the NPBE, whereas it grows linearly for the LPBE. Thus the NPBE tends to suppress effects arising from high concentrations of electric charge. This can be especially important near the active sites of certain enzymes and, indeed, it has been found that reduced reaction rates are obtained for diffusion-controlled catalysis by superoxide dismutase[21] when the NPBE is used instead of the LPBE.

These considerations make the development of an efficient method for solving the full NPBE a worthwhile goal and various methods have been proposed to circumvent the problems associated with the exponential nonlinearity in (1). One approach, which has been applied in a variety of systems, has been to represent the hyperbolic sine by its first three Taylor series terms, resulting in a fifth-degree polynomial.[17,22,25] Although this represents an improvement over the LPBE, it still suffers from the same deficiencies and again gives the improper scaling behavior with charge. Another method involves building up a solution to the full NPBE by solving a sequence of auxiliary equations in which $\sinh u$ is represented by more and more terms of its power series.[21] This has been used in an investigation of the effects of nonlinearities in the diffusion-controlled reaction between O_2^- and superoxide dismutase. This technique is extremely time-consuming versus the linear solution and for highly nonlinear problems, an increasing amount of effort is required to obtain a fully converged solution. In an alternate procedure, a variational formulation of the NPBE is used along with optimization techniques to find the minimum of an associated energy integral, which corresponds to the solution of the NPBE.[23] This approach, which is a special case of the Fletcher–Reeves algorithm,[27] has so far been applied only to problems with spherical symmetry. More recently, an application of multigrid techniques to the NPBE has been pursued by Oberoi and Allewell[26] in an investigation of lysozyme titration curves.

We have previously presented a multigrid, or multilevel, method for numerical solution of the LPBE[28] and showed that for two test problems, the multilevel method was superior to relaxation and Krylov subspace methods that had previously been dis-

cussed in the literature,[14] as well as to some improved versions of the existing methods that had not been previously applied to the LPBE. It was demonstrated that the method becomes more advantageous as the problem size grows, which will become increasingly important as more accurate numerical solutions, and hence larger discrete problem sizes, are required. In this paper, we present a hybrid method in which a robust nonlinear Newton iteration scheme is combined with a rapid linear multilevel algorithm in order to obtain a solution of the nonlinear system of equations arising from discretization of the full NPBE. The form of the Newton algorithm that is employed is globally convergent and yields superlinear convergence as the trial solution becomes sufficiently close to the true solution. We show that the hybrid algorithm is superior to other methods for the NPBE, including a more standard nonlinear multigrid implementation and as in the linear multilevel case, this superiority grows with the problem size. In fact, it is observed that this nonlinear method is so efficient that it solves the full nonlinear problem in less time than some of the best existing linear methods take to solve only the linearized problem. In addition, it is the only method robust enough to solve highly nonlinear cases.

The remainder of the paper is structured as follows. We first review multilevel methods for solving linear elliptic equations and describe inexact Newton and damped Newton methods for nonlinear equations. Our algorithm, which is employed in subsequent computations, is based on a combination of these techniques. The NPBE is then solved for a variety of proteins including crambin, BPTI, the Fv fragment of HyHEL-5, superoxide dismutase, the rhinovirus coat protein, and tryptophan synthase. The present approach is shown to give far superior performance compared with other nonlinear methods for a weakly nonlinear sample problem. The superlinear convergence of the algorithm is demonstrated. It is also shown that the time required for a solution is proportional to the number of unknowns and that this time is less than that necessary for a conjugate gradient solution of the corresponding linearized equation. The scaling of the potential with respect to charge is investigated for a spherical molecule with the LPBE, NPBE, and the fifth-order truncation of the NPBE. The effect of ionic strength on the validity of the various approximations is discussed. One of the most effective tests of models for protein electrostatics is provided by Brownian dynamics simulations of diffusion-controlled reactions. The effect of nonlinearities in these studies is explored in this paper, where simulations are run for both superoxide dismutase catalysis and a model antibody–antigen system. Finally, the power of the multilevel approach for solving large systems is demonstrated by an application of the method to the tryptophan synthase dimer, one of the largest proteins for which a crystal structure exists. Contour maps for the electrostatic potential from both the LPBE and the NPBE are shown. We conclude with a brief summary of our results.

MATERIALS AND METHODS

Numerical Methods for Nonlinear Elliptic Equations

In an earlier paper,[28] multilevel methods were designed and applied to the LPBE with very good results; the methods were shown to be superior to highly competitive techniques such as preconditioned conjugate gradient algorithms and successive overrelaxation. Multigrid methods have additionally been shown to be extremely effective for a wide range of problems and are provably of *optimal order complexity* for certain classes of problems,[29–31] in that the work required to solve for N unknowns is $O(N)$. This means that the advantages of multigrid over other methods will increase with the size of the problem, since no other methods are known to share this optimal complexity property. Here we wish to exploit some of these previously developed techniques[28] to the more numerically challenging NPBE. Although nonlinear generalizations of multilevel algorithms are available in the literature,[32–37] we often encountered difficulties in obtaining convergence when these methods were applied to the NPBE for various proteins. We thus adopted a different approach in which the advantages of the linear multigrid algorithm are combined with more robust nonlinear methods.

The nonlinear solver to be used below employs a variant of the multidimensional Newton's method. At each Newton iteration, a solution to a system of linear equations is required. This will be handled by linear multigrid techniques, which enables us to retain the speed and $O(N)$ convergence associated with the method. The additional use of the nonlinear Newton solver, however, enables us to include features which increase both the efficiency and robustness of the algorithm. One such improvement, referred as an inexact Newton method, solves the linear system only approximately during the first few iterations. Another enhancement involves the modification of the iterative equations by judiciously chosen damping parameters. The technique to be discussed below combines all of the above ideas and will be referred to as the multigrid damped inexact Newton (MUGDIN) algorithm. The method yields solutions of the NPBE faster than all previously used techniques, except multigrid, for the LPBE and, more importantly, has provided converged results for all of the cases that we have so far examined.

Linear Multigrid Methods

Multigrid methods are efficient numerical algorithms for solving the algebraic equations which

arise from discretizations of partial differential equations. We refer to our study of the LPBE[28] or to the standard references[31,38] for discussions of the basic correction scheme algorithm for linear elliptic equations, as well as examples of prolongation, restriction, and smoothing operators, and the distinctions between the V-Cycle, W-Cycle, and the FMG algorithms. An elementary introduction to the subject is given by Briggs.[39]

In a standard single-grid algorithm, an iterative scheme is devised to solve the matrix equation which arises from a discrete (finite difference, finite volume, or finite element) approximation. That is, an initial guess is made for the solution and the error to this guess is reduced through the repeated application of an equation based on the original matrix. Although this procedure is quite efficient at smoothing out the error, i.e., eliminating the portions of the error which have length scales on the order of the grid spacing, it is much less successful in handling the lower frequency components of the error. This is because the discrete approximation, for local operators allows for efficient communication only between neighboring nodes. Multilevel algorithms circumvent this problem by simultaneously discretizing the partial differential equation on grids of several resolutions. In some earlier studies of the LPBE, a multiple-grid approach was employed to obtain an improved initial guess prior to iteration.[40] In the algorithm presented here, the various grids are also used to accelerate the convergence of the iterative process.

We outline the multilevel method for a linear elliptic equation $Lu = f$ as this algorithm will be employed in our solution to the full nonlinear equation. We define a sequence of discrete problems, $L_{h_i}u_{h_i} = f_{h_i}$, by some discretization method, which we number $h_1, \ldots, h_k$ with $h = h_k$ denoting the finest grid and $H = h_{k-1}$ denoting the next finest grid. We can furthermore define linear *restriction* and *prolongation* operators R and P, respectively, which map grid function from the fine grid to the coarse grid, and from the coarse grid to the fine, respectively. For certain discretizations, such as the finite element method, it is automatically the case that

$$L_H = RL_hP \tag{3}$$

where R and P are chosen appropriately. In the finite-difference computations presented here, the various L_{h_i} are constructed by directly discretizing the partial differential equation on each grid. This method is suitable provided that the discontinuity in the dielectric constant is not too large[9] as is the case for protein electrostatics. Alternatively, the matrix L_H can be obtained algebraically from L_h by directly using (3). This procedure is computationally more intensive, but may be possibly advantageous from a theoretical standpoint.[31]

The linear multigrid method[31] or correction scheme[38] for solution of the discretized equations

$$L_hu_h = f_h \tag{4}$$

is based on the *error equation;* at iteration j the method solves for the *error* $e_h^j = u_h - u_h^j$ from

$$L_he_h^j = L_h(u_h - u_h^j) =$$
$$L_hu_h - L_hu_h^j = f_h - L_hu_h^j = r_h^j \tag{5}$$

where r_h^j is the *residual.* If the error is smooth compared to the scale of the grid spacing, it can be equally well represented on a coarser grid. Therefore, a *smoothing* method, such as Gauss–Seidel or Jacobi iteration,[41] which efficiently damps out the high-frequency components of the error is performed on the fine grid. In such an approach, L_h is split into $L_h = M_h - N_h$. If M_h is taken to be the diagonal part of L_h, this procedure yields the Jacobi method, while if M_h is the lower triangular part of L_h, Gauss–Seidel iteration is obtained. The solution to (4) obeys the equation

$$u_h = M_h^{-1}N_hu_h + M_h^{-1}f_h \tag{6}$$

which is used to generate the iterative scheme

$$u_h^{j+1} = S_hu_h^j + T_hf_h \tag{7}$$

where $S_h = M_h^{-1}N_h$ *and* $T_h = M_h^{-1}$. After the above smoothing method, which is chosen to be Gauss–Seidel iteration for the present algorithm, is performed on the fine grid, the error equation is then solved on a coarse grid. This result is transferred back to the fine grid through the use of the prolongation operator P. Finally, the smoothing method can again be applied to the corrected solution.

A simple two-grid version of this iteration can be formulated as

Let u_h^0 be an initial approximation.
Do $j = 0,1,2, \ldots$ until convergence:
1. Presmooth: $\bar{u}_h = S_hu_h^j + T_hf_h$
2. Restrict residual: $r_H = R(f_h - L_h\bar{u}_h)$
3. Solve for correction: $e_H = L_H^{-1}r_H$
4. Prolongate and correct: $\bar{\bar{u}}_h = \bar{u}_h + Pe_H$
5. Postsmooth: $u_h^{j+1} = S_h\bar{\bar{u}}_h + T_hf_h$
End Do. (8)

It can be seen that step 3 requires the solution of a linear equation on the coarser grid. Thus, the error to this equation can be corrected on the next coarsest grid in the same manner as above. One can continue to recursively descend to the coarsest grid h_1 where, if the initial grid size is properly specified, the linear equation can be effectively solved using direct matrix inversion. Some standard approaches for cycling between fine and coarse grids are depicted graphically in Figure 2; these algorithms are described in more detail elsewhere.[28,31]

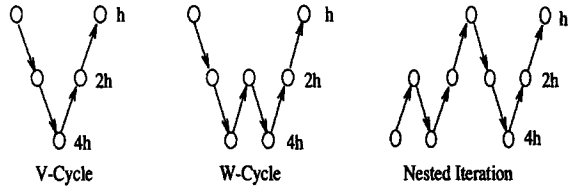

Fig. 2. Nested sequence of meshes in three common multigrid algorithms. The leftmost diagram shows the V-cycle which involves correcting with the coarse grid once at each iteration. The W-cycle entails two coarse grid corrections per level at each iteration. The final diagram illustrates a boot-strapping technique which provides improved initial approximations.

Newton-Multigrid Iteration

We first remark that the question of uniqueness of solution is a nontrivial matter, especially in the nonlinear case. For example the nonlinear Bratu problem,[42] which is quite similar to the NPBE, possesses nonunique solutions in certain cases. It has been shown,[9] however, that both the NPBE and the nonlinear algebraic equations arising from a standard finite volume discretization are uniquely solvable for any molecule defining the right-hand side function f_h.

The method adopted for solving the discretized approximation to the nonlinear problem involves a generalization of Newton's method for finding the roots of a one-dimensional function.[20] In this simpler case, the function $a(x)$ is approximated by the first two terms of its Taylor series leading to

$$a(x) \approx a(x^0) + (x - x^0)a'(x^0) = 0 \qquad (9)$$

where x^0 is an initial guess to the solution. This yields the iteration

$$x^{j+1} = x^j - a(x^j)/a'(x^j). \qquad (10)$$

This equation has a simple geometrical interpretation; the updated guess x^{j+1} is where the tangent line of $a(x)$ at x^j intersects with the x-axis. There may, however, be instances when $a'(x^j)$ becomes very small, leading to divergence of the second term in the above equation. In practice, this implies that (10) is not guaranteed to converge from an arbitrary initial guess. Thus *damping parameters* ω_j are introduced such that

$$x^{j+1} = x^j - w_j a(x^j)/a'(x^j) \qquad (11)$$

in order to ensure convergence of the algorithm. Of course, the ω_j must go to one as x^j approaches the solution.

Now, let $N_h(u_h) = f_h$ denote the system of n nonlinear algebraic equations corresponding to a discretization of the second-order nonlinear elliptic partial differential equation $N(u) = f$, and denote the derivative, or Jacobian matrix, of N_h at u as $DN_h(u)$. Since we wish to solve the equation $N_h(u_h) - f_h = 0$, the appropriate analog to the above procedure is as follows:

$$\left\{\begin{array}{l} \text{Let } u_h^0 \text{ be an initial approximation.} \\ \text{Do } j = 0,1,2,\ldots \text{ until convergence.} \\ \quad 1.\ \text{Solve linear system for} \\ \qquad \text{correction:} \quad DN_h(u_h^j)\,\delta_h^j = f_h - N_h(u_h^j) \\ \quad 2.\ \text{Correct the solution:} \quad u_h^{j+1} = u_h^j + \omega_j \delta_h^j \\ \text{End Do.} \end{array}\right\} \qquad (12)$$

The vector δ_h^j defines a direction in multidimensional space along which to find a correction to the solution and the damping parameters ω_j increase the robustness of the algorithm. In fact, it can be shown[20] that the Newton direction δ_h^j is always a *descent direction,* in the sense that for each Newton direction δ_h^j, there exists some steplength ω_j such that

$$\|N_h(u_h + \omega_j \delta_h^j) - f_h\| \leq \|N_h(u_h) - f_h\|. \qquad (13)$$

Therefore, we perform one-dimensional line searches along the Newton direction δ_h^j until (13) is satisfied. In this sense, the method is globally convergent.

Step 1 of iteration (12), which requires the solution of a linear system of equations, can be efficiently implemented with the linear multigrid method discussed above. An advantage to the above method is that if the approximation u_h^j is close enough to the solution u_h, then the Newton's method will converge superlinearly or even quadratically, although the convergence rate of the entire iteration will be dominated asymptotically by the linear convergence of the multigrid method.[31,43] Such a Newton-multigrid iteration has been used successfully for other applications.[43,44]

Another modification which increases the efficiency and robustness of the algorithm is motivated by the observation that, for the first few iterations when u_h^j is far from the true solution, it is hardly necessary to solve the linear system in Step 1 of (12) exactly; the only requirement is that the Newton direction be approximately correct. This is the reasoning behind the inexact Newton methods,[45–48] in which the system of linear equations is solved only within a very loose tolerance at first, with the tolerance being gradually tightened to the desired accuracy of solution as the iteration proceeds. With the appropriate tolerance criteria at each Newton iteration, it can be shown that this algorithm also convergences superlinearly.[47] An analysis of the interplay of the damping and inexact solutions of the Newton equations can yield criteria for both the damping parameters and the tolerances to which the linear equations are solved which guarantee a global superlinear convergence of the Newton iteration.[46]

The solutions of the nonlinear Poisson–Boltzmann equation to be presented here employ the Newton

iteration given in (12) with linear multigrid, designed specifically for the linearized Poisson–Boltzmann operator,[49] used as an inexact solver in Step 1 and the damping parameter in Step 2 chosen to provide global convergence properties. As noted above, this hybrid procedure is referred to here as the MUGDIN algorithm.

RESULTS

Performance of Algorithm

Calculations are performed below for a variety of proteins in order to test the present algorithm within the context of applications to biological systems. Since the MUGDIN method is based on a solution of the system of equations which results from the finite-difference approximation to the Poisson–Boltzmann equation, it can be incorporated into any program which employs such a formalism. We have implemented our code as a module within the UHBD program,[6] which sets up the problem for the multigrid solver.

The following details hold for each of the calculations described in this section. All of the atomic coordinates are taken from the Brookhaven Protein Data bank. The molecular volume is obtained by identifying radii from the OPLS force field[50] with each of the atoms. A residue-based model is used to define the charge distribution of the protein by assigning appropriate unit charges to the following atoms: CG of Asp, CD of Glu, NZ of Lys, and CZ of Arg. Where necessary, charges are also given to the N and C atoms of the N- and C-termini, respectively, and to CE1 of His. We note that the multigrid algorithm works just as well for more detailed atomic charge models, as will be shown in the next section. No Stern layer is used and this tends to emphasize the magnitude of nonlinear effects, since the ion density is nonzero up to the surface of the molecule where the potential tends to be strongest. The dielectric constant inside the molecule is assumed to be 2 and the dielectric constant of the solvent taken as 78. At the protein–solvent boundary, a smoothing procedure is employed in which the dielectric constant is assigned an intermediate value according to how close a grid point is to the interface.[6] All calculations are done at a temperature of 300 K and are deemed to be converged when the norm of the residual vector r_h divided by the norm of f_h becomes less than 10^{-8}. For the conjugate gradient methods, the convergence criterion is that the norm of $(u_h^{i+1} - u_h^i)$ be less than 10^{-8} times the norm of u_h^i. The timings are obtained by running the program in double precision on one vector processor of a Convex C240 for crambin and that of a Convex C3880 for all other cases.

The boundary condition $g(r)$ for both the LPBE and NPBE, as defined in (2), is obtained by summing the potentials of all of the charged atoms as if each atom is an independent Debye–Hückel sphere of radius 2 Å. This assumes the validity of a linear approximation on $\partial\Omega$. As long as the boundary is chosen sufficiently far away from the surface of the molecule, the magnitude of u will be small and this should be a good approximation. Ultimately, however, it will be best to choose the boundary conditions so as to properly reflect the nonlinear nature of the equation. One possible way to achieve this would be to construct look-up tables based on the solution of the NPBE for Debye–Hückel spheres of various charges and radii. The construction of $g(r)$ is potentially very time-consuming, particularly when charges are assigned to each atom, as it requires on the order of $l^2 N_m$ operations, where l is the number of nodes in each direction and N_m is the number of charges. In order to present a more clean comparison between various calculations, each CPU time reported on below is that which is required only to solve the appropriate finite-difference equation and does not include any of the set-up time associated with the problem.

Comparison With Other Methods

The exponentials that occur in the hyperbolic sine function of the NPBE are a source of great difficulty for nonlinear methods. For most of the cases discussed here, standard nonlinear solvers were unable to achieve converged solutions to the NPBE. We therefore show in Figure 3 a comparison of various methods using a small, mildly nonlinear problem. The electrostatic potential for the 46-residue protein crambin,[51] with a total of six charges and a net charge of zero, is solved for on a 31^3 grid, with 2 Å spacing at an ionic strength of 1 mM. Although the grid is quite small for the molecule being considered and the salt concentration is very low from an experimental standpoint, this example is indicative of the conditions that are necessary in order to obtain converged solutions to the NPBE for methods other than the one described here. We note that we have had better success with other methods when the fifth-order polynomial approximation to the nonlinearity is employed.

The MUGDIN algorithm yields by far the most rapid convergence and is considerably faster than a more straightforward generalization of the multigrid method similar to that used by Oberoi and Allewell.[26] Next after the multigrid-based methods are the nonlinear conjugate gradient method, followed by the nonlinear successive over-relaxation algorithm. Each of these is at least three times as fast as the nonlinear Gauss–Seidel solver which was employed in a previous study of the NPBE, as described earlier.[21]

We note that it is certainly possible to devise other schemes for handling the NPBE. For example, one might use the Newton's method described in the preceding section in conjunction with some other linear

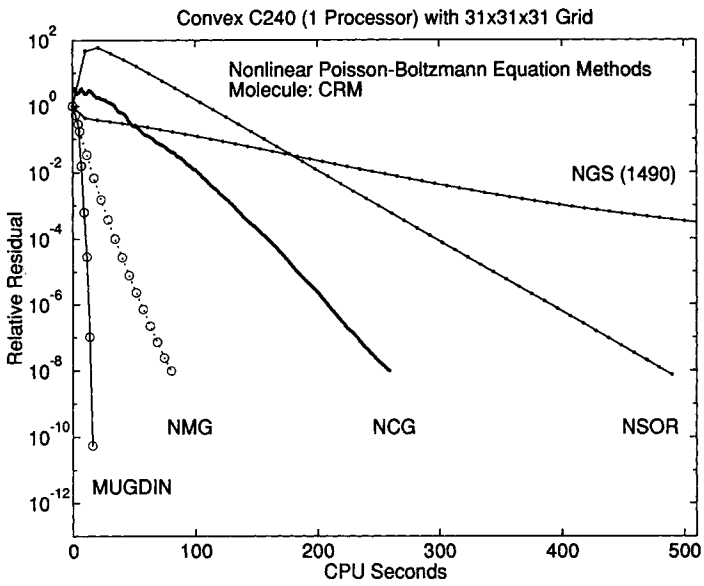

Fig. 3. Performance for various nonlinear solvers of the full Poisson–Boltzmann equation for the protein crambin at 1 mM ionic strength. The norm of the relative residual, which is a measure of the amount that the approximate solution differs from the true solution, is plotted as a function of iteration number and CPU time. The curve labeled MUGDIN is obtained using the algorithm described in the text. The other curves result from the application of straightforward nonlinear generalizations of standard linear solvers including multigrid (NMG), conjugate gradient (NCG), successive overrelaxation (NSOR), and Gauss–Seidel (NGS).

equation solver, such as a conjugate gradient method or the Gauss–Seidel iteration. However, given the advantages of the linear multigrid method for handling such systems, it is likely that such approaches will yield inferior performance against the MUGDIN solver. Interestingly, we have found that our algorithm runs faster with the full nonlinearity of the NPBE compared to its fifth-order polynomial approximation. This is due to the fact that the hyperbolic sine is a machine-level intrinsic function, while the polynomial is evaluated in higher level code. It does illustrate, though, the success of the present algorithm at handling an exponential nonlinearity.

Scaling and Convergence Properties

The remainder of the computations performed in this section are done at 5 mM ionic strength with a grid spacing of 1 Å. The proteins and grid sizes considered below are bovine pancreatic trypsin inhibitor (BPTI)[52] with 58 residues at a grid size of 63^3, the Fv fragment of the HyHEL-5 antibody[53] with 221 residues at 95^3, the superoxide dismutase (SOD) dimer[54,55] with 304 residues at 127^3, the rhinovirus coat protein monomer[56] with 804 residues at 159^3, and the tryptophan synthase dimer[57] with 1266 residues at 191^3. The properties of the above systems have been collected in Table I. The Fv fragment has zero total charge has no center of charge and in this instance, the center of the grid is chosen according to the requirement that the diagonal moments of the quadrupole tensor vanish with respect to this point.[58] In all other cases, the grid is centered at the molecular center of charge. When residues are absent from the crystallographic structures due to disorder, they are omitted from the calculation. These missing residues only occur at the N-terminal ends of some chains in the rhinovirus protein and at the end of a few helices and strands in tryptophan synthase.

In the previous section, we have discussed the superlinear convergence properties of the MUGDIN algorithm when the trial solution becomes sufficiently close to the true solution. This behavior is displayed in Table II for BPTI as well as the rhinovirus coat protein. In each case, the relative residual decreases dramatically during the last few iterations. Thus the inexact Newton's method becomes more advantageous the greater the accuracy demanded of the solution. In comparison, the norm of the residual as a function of iteration number for the multigrid solution of the corresponding LPBE shows linear convergence. An inspection of Figure 3 reveals that the straightforward nonlinear generalization of the multigrid algorithm also yields linear convergence. We have also solved the LPBE for these systems using the linearly convergent diagonally scaled conjugate gradient method which required 294 iterations for BPTI and 694 iterations for the rhinovirus protein, although each of these iterations is less expensive in terms of CPU time than a multigrid iteration.

As with the multigrid method, the MUGDIN approach is of optimal order, meaning that the work

TABLE I. Sample Proteins*

	Residues	Grid nodes	CPU time
Bovine pancreatic trypsin inhibitor	58	63 × 63 × 63	65
HyHEL-5 antibody (Fv fragment only)	221	95 × 95 × 95	235
Superoxide dismutase (homodimer)	304	127 × 127 × 127	634
Rhinovirus coat (monomer)	804	159 × 159 × 159	1220
Tryptophan synthase (homodimer)	1266	191 × 191 × 191	1690

*Sample proteins used in the present study. The first column gives the total number of residues, the second column shows the total number of grid nodes used for discretization on the finest grids, and the third column gives the CPU seconds required for a solution of the full nonlinear Poisson–Boltzmann equation at 5 mM ionic strength by the multigrid damped inexact Newton algorithm described herein.

TABLE II. Convergence Properties of Multilvel-Based Algorithms*

	Norm of relative residual			
	BPTI		Rhinovirus	
Iteration	LMG	MUGDIN	LMG	MUGDIN
0	1.00E+00	1.00E+00	1.00E+00	1.00E+00
1	1.01E−01	5.30E−01	2.80E−01	7.93E−01
2	6.88E−03	3.81E−02	3.65E−02	5.30E−01
3	1.13E−03	4.35E−03	5.58E−03	7.72E−02
4	1.78E−04	6.62E−04	1.19E−03	1.14E−02
5	5.03E−05	5.85E−05	4.28E−04	1.83E−03
6	1.72E−05	8.84E−07	9.74E−05	2.62E−04
7	2.85E−06	3.68E−10	1.78E−05	1.55E−05
8	6.68E−07		4.18E−06	1.19E−07
9	1.15E−07		8.40E−07	1.82E−11
10	1.74E−08		2.61E−07	
11	6.04E−09		7.83E−08	
12			1.75E−08	
13			4.49E−09	

*Norm of the relative residual versus iteration number for a multigrid solution (LMG) of the linearized Poisson–Boltzmann equation and a multigrid damped inexact Newton (MUGDIN) solution of the full nonlinear Poisson–Boltzmann equation. The two sample proteins for which results have been displayed are bovine pancreatic trypsin inhibitor and the rhinovirus coat protein. The LMG method displays linear convergence as a function of iteration number while the MUGDIN algorithm yields superlinear convergence during the last few iterations when the trial solution is sufficiently close to the true solution.

needed to solve for N unknowns is proportional to N. This desirable property is shown in Figure 4, where the CPU times required to solve the LPBE and NPBE with several methods for various proteins are plotted. It is seen that while the multigrid solution of the LPBE is faster than the MUGDIN method for the NPBE, the latter is still much more efficient than the diagonally scaled conjugate gradient method for the LPBE by as much as a factor of three. Since the nonlinear solver displays optimal order behavior, the advantages are greater the larger the problem size. Thus, not only does the present algorithm provide for a full consideration of nonlinear effects implied by the Poisson–Boltzmann equation, it also allows for the treatment of large systems which have heretofore been computationally intractable.

Scaling With Respect to Charge

From a simple inspection of the NPBE, it can be inferred that the electrostatic potential increases only logarithmically with charge as opposed to linearly with the LPBE. As noted previously, this feature of the NPBE significantly reduces the biological consequences of highly charged molecules. To investigate this property, we have performed calculations for spherical molecules at various charges and ion concentrations. Shown in Figure 5 is the potential 2 Å away from the surface of a sphere of 15 Å radius as a function of charge at two ionic strengths. A numerical solution of the full NPBE is given as well as the analytical solution of the LPBE. For comparison, the potential has also been computed using a fifth-order polynomial approximation for the hyperbolic sine function. For the nonlinear equations, the imposed boundary condition is that the solution approach the LPBE potential at the outer faces of the grid. A very large grid size, 191 nodes on a side, was employed to minimize the effects of this approximation, with 2 Å spacing at 30 mM ionic strength and 1 Å resolution for 150 mM.

The logarithmic scaling of the NPBE can be

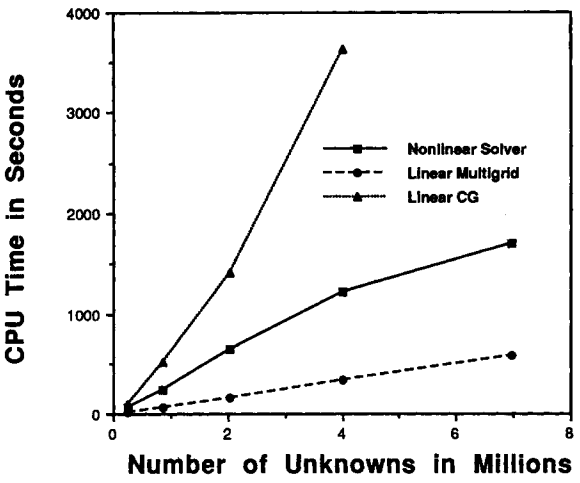

Fig. 4. CPU time as a function of the number of unknowns for solution of the linear and nonlinear Poisson–Boltzmann equations. The solid line gives the MUGDIN solution time, while the other curves represent the time required for the multigrid and diagonally scaled conjugate gradient methods to solve the linearized Poisson–Boltzmann equation. The sample proteins are listed in Table I and all calculations are done at 5 mM.

clearly observed in the graphs. It is also seen that the fifth-order result is often a very reasonable approximation, particularly for lower charge and higher ionic strengths. There are regimes, however, where this representation breaks down as the potential asymptotically scales as the one-fifth power of the charge rather than the logarithm. Although in some cases, the charge of the sphere is much larger than what would be expected for proteins, the study is meant to be merely illustrative and we have indeed found the same qualitative behavior in biological systems, as has been explicitly discussed for the case of SOD.[59]

Electrostatic Steering

An important application of the Poisson–Boltzmann equation is to Brownian dynamics simulations of diffusion-controlled biomolecular reactions.[60] In some systems, it has been found that the rates for such reactions are enhanced through long-range forces, as the two molecules can mutually guide each other into a configuration favorable for reaction. The interactions which are responsible for the steering are electrostatic in nature and can be determined through solution of a Poisson–Boltzmann equation. The NPBE is particularly relevant in this regard as appreciable steering occurs primarily when u approaches unity, which is precisely the regime where the LPBE breaks down.

A comparative study by Allison, Sines, and Wierzbicki[21] has been carried out for simulations of SOD catalysis with both the LPBE and NPBE. It was found that significantly reduced reaction rates can occur when the NPBE is used, with the amount of reduction dependent on the ionic strength. These results can be understood in terms of the contour maps that we have calculated for SOD,[59] where the potential in the vicinity of the active sites can be seen to be much larger with the LPBE than for the NPBE. This is a specific example of the increased damping of the field due to the protein that occurs in the NPBE, since the mobile ions are allowed to react to external fields beyond a linear response approximation. This is of course related to the different scaling behaviors with respect to charge of the two equations.

The Brownian dynamics simulations reported on in this section are carried out with the UHBD code.[6] In these computations, one molecule is held fixed and rigid at the center of two concentric spheres while the other (in both cases below, a charged sphere) undergoes diffusional motion under the influence of the field of the first. The second molecule is initially placed on the inner sphere and its motion is propagated in discrete time steps Δt according to the equation[61]

$$\Delta \mathbf{r} = \frac{D}{k_B T} \mathbf{F} \Delta t + \mathbf{R} \tag{14}$$

where $\Delta \mathbf{r}$ is the change in position, D is the relative diffusion constant, and $\mathbf{R}$ is a random vector satisfying certain statistical constraints. The quantity $\mathbf{F}$ gives the electrostatic force obtained from the Poisson–Boltzmann solution for the first molecule as $\mathbf{F} = -\nabla\Phi$. The motion is continued until either a predetermined reaction condition is met or the outer sphere is reached. The reaction rate k, which is determined statistically from a large sample of trajectories, can be related to the probability β that such a trajectory will yield a reaction by the equation[62]

$$k = \frac{k_s(b)\beta}{1 - (1 - \beta)k_s(b)/k_s(q)} \tag{15}$$

where b and q are the radii of the inner and outer spheres, respectively, and $k_s(a)$ is the analytical result for the reaction rate for diffusion to a sphere of radius a in a spherically symmetric potential $U(r)$ which obeys the expression

$$k_s(a) = 4\pi D \left[\int_a^\infty \frac{\exp[U(r)/kT]}{r^2} dr \right]^{-1}. \tag{16}$$

We note that in order to be consistent with the use of the NPBE, one should employ the nonlinear solution in computing the above integral. For the antibody-antigen system below, the antibody is uncharged and thus $U(r) = 0$ in (16). For SOD, $U(r)$ is sufficiently small in the range of integration that essentially the same results are obtained whether the linear or nonlinear expression for $U(r)$ is used. Hence, all of the differences reported below arise in the

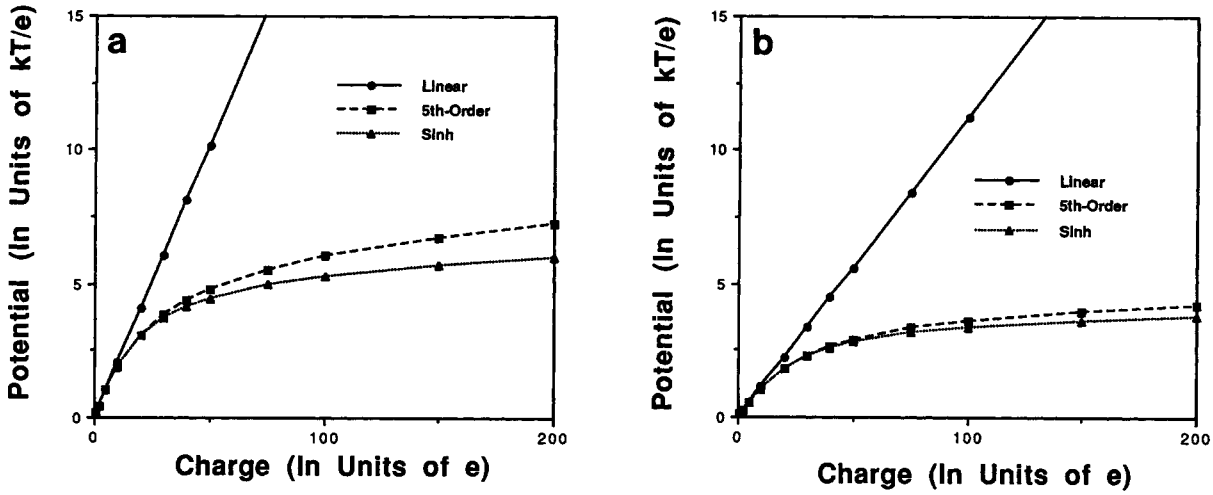

Fig. 5. The electrostatic potential at a distance of 2 Å away from the surface of a 15 Å spherical molecule at (a) 30 mM and (b) 150 mM ionic strength. The graphs compare the solutions of linearized, fifth-order, and full nonlinear Poisson–Boltzmann equations.

reaction probability β as obtained from the Brownian trajectories.

Antibody–Antigen System

Both experimental and theoretical evidence suggests that the rate for antibody–antigen association is limited by diffusion.[63,64] A detailed study of electrostatic steering in such a system has been undertaken through the use of Brownian dynamics.[58] The calculations were based on the X-ray crystal structure for the monoclonal antibody HyHEL-5 bound to hen-egg lysozyme.[53] The lysozyme carries a large net positive charge and the complex is stabilized by three salt bridges involving two charged residues from each of the proteins. Experimentally, it has been found that mutations of these important residues have a drastic effect on antibody–antigen binding.[65,66] Here some of the same simulations are repeated with the exception that the NPBE is used in place of the LPBE. The Fv fragment of the antibody plays the role of the fixed molecule in the above description while the lysozyme is approximated as a positively charged sphere. The NPBE for the Fv fragment is solved on a 95^3 grid exactly as described in the preceding section, except that the ionic strength is varied from 5 to 350 mM. In all other aspects, the calculations are performed as described previously.[58]

The simulations are run for the wild-type Fv fragment at several ionic strengths and with three different lysozyme charges. The ratios of the reaction rates for the NPBE solution to the LPBE solution are presented in Table III. As has been found in the SOD simulations, the use of the NPBE has resulted in a decrease of the reaction rate in all cases. Again this is presumably due to the increased electrostatic screening by the ions which is reflected by the nonlinear terms. The ratios vary in an irregular fashion with both salt concentration and charge and the reaction rate may be suppressed by as a much as an order of magnitude. The difference between the linear and nonlinear cases also appears to be getting smaller at lower lysozyme charge and higher ionic strength where electrostatic steering is less important. It is clear that use of the NPBE may affect the ionic strength dependence of the reaction rates calculated through Brownian dynamics. Moreover, the unpredictable manner in which the reaction probability is altered as a function of charge provides indications that the LPBE may be unreliable for computing *relative* reaction rates between various mutants of a system.

Superoxide Dismutase

There is an extensive history of Brownian dynamics simulations for SOD catalysis as it has been a prototype for the study of electrostatic steering in biomolecular reactions. Early researchers employed simple charge distributions[67] to simulate the field of superoxide dismutase. More sophisticated atomic charge models,[68–71] along with the LPBE, have been used to determine the effects of point mutations on the reaction rate.[72,73] Getzoff et al.[74] have performed computations in which reaction rates for wild-type human SOD catalysis are compared with various mutants of the protein. Although most of the calculated relative rates are in reasonable agreement with experiment, there are some quantitative differences particularly concerning one of the double mutants, which has been attributed to structural changes in the enzyme. As noted above, Allison et al.[21] have found that the use of the NPBE can have

TABLE III. Supression of Reaction Rates in Brownian Dynamics Simulations*

Lysozyme charge ionic strength in mM	(Nonlinear Rate/linear rate)		
	5e	3e	1e
5	0.62 ± 0.02	0.34 ± 0.04	0.45 ± 0.12
25	0.34 ± 0.04	0.19 ± 0.08	0.47 ± 0.13
75	0.16 ± 0.07	0.24 ± 0.10	0.59 ± 0.15
150	0.22 ± 0.09	0.45 ± 0.12	0.66 ± 0.16
350	0.56 ± 0.12	0.67 ± 0.17	0.69 ± 0.16

*Ratio of reaction rates for a model antibody–antigen system using the linear and nonlinear Poisson–Boltzmann equation to determine the extent of electrostatic steering. The ratio gives the reaction rate obtained with the nonlinear equation over the reaction rate implied by the linear equation. The estimated errors are statistical.

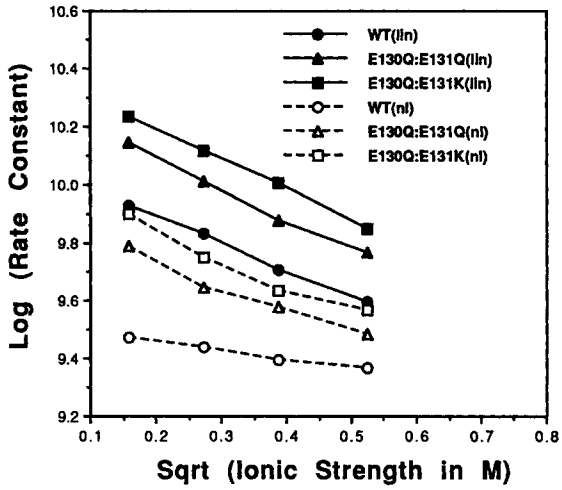

Fig. 6. Rate constants as a function of ionic concentration obtained from Brownian dynamics simulations of catalysis of superoxide by wild-type bovine superoxide dismutase and two mutants. The solid curves result from using the linearized Poisson–Boltzmann equation to determine the electrostatic steering, while the dashed curves give the rate constants when the full nonlinear equation is used.

a considerable effect in the SOD system. We have therefore performed our own simulations with the NPBE in order to reexamine the results of Getzoff et al.[74]

Because the atomic coordinates for human SOD are not yet generally available, the present investigation is performed with bovine SOD rather than human SOD. Otherwise, we have attempted to keep as close to the above calculation as possible. Thus we have added hydrogens to the structure and employed an all-atom charge model based on the OPLS force field. The reduced SOD form is considered in which the copper ions are given a charge of $+1e$ rather than $+2e$, with the extra charge being shifted to the His-61 residues. From the simulations of Sines et al.[72] and our own studies, it has been found that this slight change has a negligible effect on the reaction rate. The boundary condition on $\partial\Omega$ is enforced by assuming that $g(r)$ is generated by a single Debye–Hückel sphere with the total enzyme charge and a radius of 30 Å. The parameters associated with the diffusional trajectories are kept identical to those of Getzoff et al.,[74] who also utilized the UHBD program except that in the present case only 10,000 trajectories are run for each mutant.

We have run simulations for the wild-type bovine SOD and the two double mutants analogous to those of Getzoff et al.[74] Each double mutant involves the modification of two consecutive Glu residues near the active site and with the bovine protein these are given by A: (Glu-130→Gln,Glu-131→Gln) and B: (Glu-130→Gln,Glu-131→Lys). Reaction rates for simulations with both the LPBE and NPBE are presented in Figure 6. The relative rates and ionic strength dependence for the LPBE are very similar to those obtained for human SOD. One feature of the NPBE simulations is that the plots of the logarithm of the rate versus $\sqrt{I}$ are less linear for the NPBE simulations. As in the previous system, the NPBE rates are again lower, but the relative rates are qualitatively unchanged, whereas experiment indicates that the reaction rates for the analogous mutation to mutant B are less than those for the analog to mutant A. Given the similarity between human and bovine SOD, it is likely that this conclusion will hold for both cases. Thus it appears that the use of the NPBE does not resolve the discrepancy between theory and experiment in this instance. It has been suggested that the difference lies in structural changes in the enzyme[74] and such questions are more appropriately addressed within the framework of molecular dynamics.[75] We finally note that as the total electrostatic potential of the molecule decreases, the differences between the LPBE and NPBE calculations will become smaller, although the use of the NPBE is always to be preferred for Brownian dynamics simulations since it represents a more physically realistic model.

Tryptophan Synthase

As a final application of the MUGDIN algorithm, we undertake an investigation of the tryptophan synthase homodimer, a protein roughly 150 Å in length, which contains 1266 residues that have been resolved to 2.5 Å through X-ray crystallography.[57] Each monomer of the complex consists of two subunits, α and β, which catalyze a two-step reaction; the α-subunit reacts with an indole 3-glycerol phosphate molecule to produce indole which is then apparently channeled through an internal tunnel to the β-subunit to react with a serine in order to form the final product, tryptophan.[76] The subunits of the molecule are arranged in a roughly linear fashion as αββα, which can be seen in Figure 7.

The calculation is done on a 191^3 grid with 1 Å

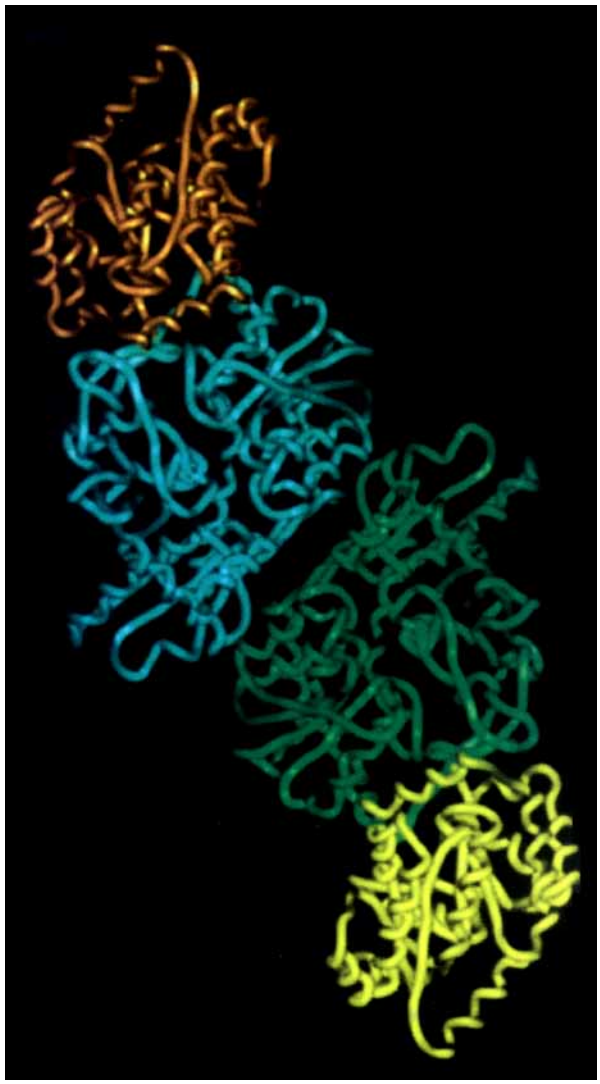

Fig. 7. Depiction of the tryptophan synthase dimer. The two identical β-subunits are flanked by the α-subunits. Each chain is shown in a different color.

spacing and all other aspects of the calculation are as outlined previously with exception that a Stern layer of 2 Å thickness is now included. As alluded to earlier, this modification tends to suppress effects arising from the nonlinearity. The His residues near the active sites of the β-subunits, B86 and B115, are assumed to be positively charged making the total charge of the system $-24e$. We note that the size of problem could possibly be reduced through the use of rectangular grids, or perhaps through focusing techniques. However, because it is wished to demonstrate the MUGDIN method for a large problem and since the calculation requires only 5 CPU minutes for the LPBE and 30 CPU minutes for the NPBE, we have followed the above formulation.

Color contour maps of the electrostatic potentials in a plane near the β active sites are shown in Figure 8, with calculations for both the LPBE and NPBE performed at 5 and 150 mM ionic strength. The excluded volume of the protein together with the Stern layer is rendered in white and the molecule has the same orientation as in Figure 7. The distinctions between the LPBE and NPBE are larger at 5 mM, but still persist at 150 mM. It is apparent that these differences are greater when the magnitude of the potential is larger, although changes in the contour lines where u is less than one can also be discerned. As expected the magnitude of the potential is always smaller for the NPBE, again due to the increased efficiency of electrostatic screening.

CONCLUSIONS

We have herein described a practical method for numerically solving the discretized version of the three-dimensional full nonlinear Poisson–Boltzmann equation. The technique combines a multilevel approach with an inexact damped Newton's method for solving nonlinear systems of equations. The algorithm scales linearly with the number of unknowns and converges superlinearly to the solution as the iteration proceeds. Our results indicate that the present algorithm is extremely robust in providing converged solutions to the NPBE in a wide range of ionic strengths for a variety of proteins. Furthermore, the algorithm yields a solution to the NPBE more rapidly than do the commonly used single-grid algorithms for the corresponding linearized equation. We are also able to test the validity of low-order polynomial representations of the nonlinearity, although use of the present method appears to make such approximations unnecessary.

The above properties have been verified for a number of proteins using grid sizes of up to 191^3 nodes, with the advantages becoming especially apparent for the largest systems. The application of the nonlinear solver to tryptophan synthase shows that it is possible to consider large molecules which otherwise might have been thought to be intractable. In attempting to compare our method with straightforward nonlinear generalizations of various linear equation solvers, it was found that the latter yielded converged solutions only for very small problems at extremely low ionic strength, due to the exponential terms that occur in the equation. We also encountered some of the same difficulties with the straightforward nonlinear multigrid technique. Thus, as of now, the present algorithm offers the most efficient and robust method for solving the full nonlinear Poisson–Boltzmann equation.

Although we have confined our investigations to monovalent electrolytes, the Poisson–Boltzmann equation, with suitable modifications, can be employed to describe polyvalent and mixed electrolytes. For these cases, the nonlinearity will be more severe than in Eq. (1) and hence the robustness of

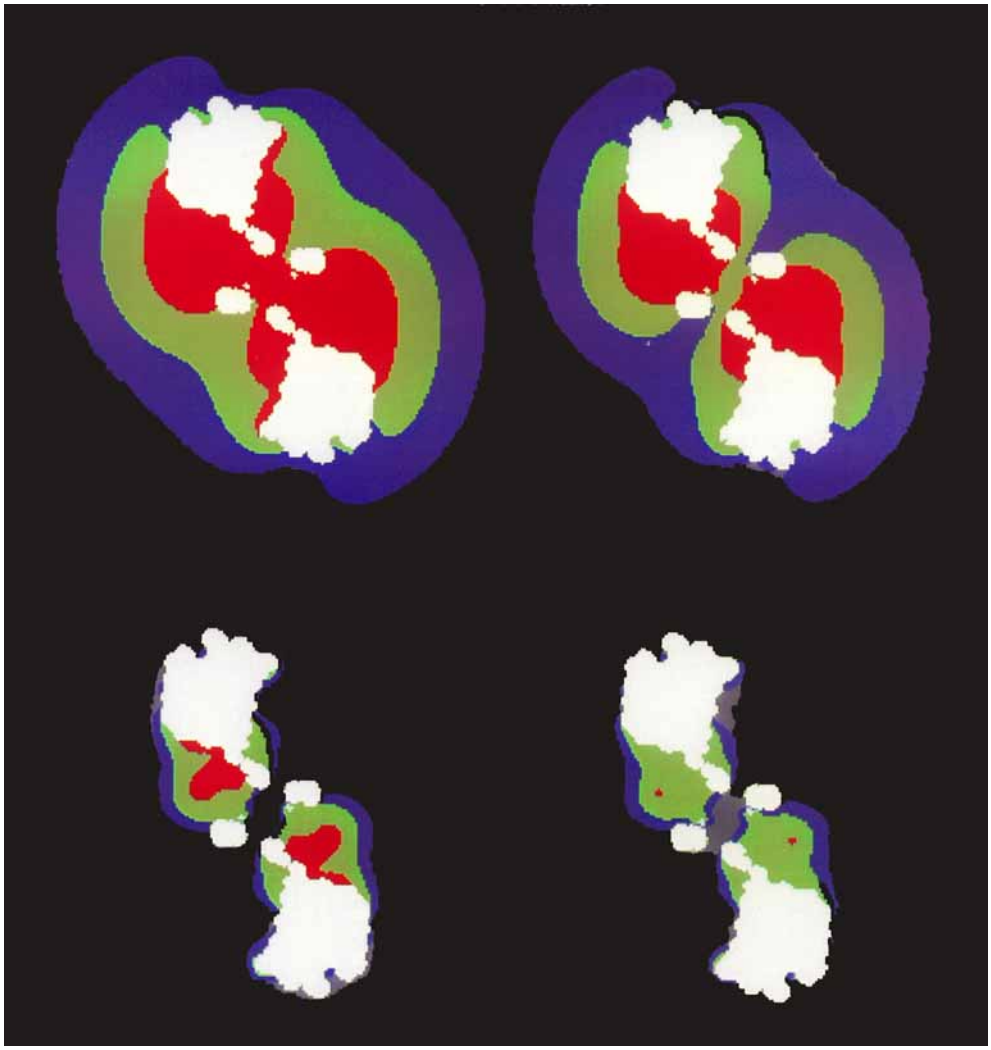

Fig. 8. Color contour maps of the electrostatic potential in a two-dimensional slice through the tryptophan synthase dimer. The top two diagrams represent solutions at 5 mM obtained from the linearized (**left**) and nonlinear (**right**) Poisson–Boltzmann equations. The bottom two pictures show maps for 150 mM, with the left and right again respectively representing linear and nonlinear solutions. The molecule is in the same orientation as in Figure 7. The excluded volume is given in white. The red portion is where the potential is larger than $2kT$, the green area is where it is between kT and $2kT$, and the border of the blue and gray areas is the $kT/2$ contour line.

the solver will take on increased importance. It should be noted, though, that such a continuum model, which treats ion–ion correlations in only an average manner, may have less validity in these situations. In fact, even for 1:1 electrolytes the Poisson–Boltzmann approach breaks down at sufficiently high ionic strengths. However, detailed atomic-level descriptions are currently difficult to implement for ionic solutions. Thus, the NPBE remains a useful and necessary tool for modeling salt effects in biological systems for a wide range of conditions.

The ability to solve the full nonlinear Poisson–Boltzmann equation implies that one can undertake investigations of biomolecular systems with a more accurate representation of ionic strength effects. In our studies, we have found that the electrostatic potential from the NPBE tends to be more strongly damped than with the LPBE because the ions are allowed to fully respond to the molecular charge. As this charge increases in magnitude, the resulting electrostatic potential grows only logarithmically with the NPBE rather than linearly as with the LPBE. This mechanism tends to mitigate the biological effects of highly charged systems and can be of importance near the active sites of enzymes where large concentrations of charge are sometimes found. As one application, we have briefly examined pos-

sible consequences on computer simulations of diffusion-controlled reactions, although such considerations will be of significance whenever continuum models of protein electrostatics are employed. It is clear that the multigrid damped inexact Newton algorithm described here can be of use in such studies.

ACKNOWLEDGMENTS

This work is based in part on a Ph.D. thesis submitted by M.H. to the University of Illinois. We are grateful to J.A. McCammon for providing us with a copy of the UHBD program. We also thank A. Nicholls for helpful discussions. We appreciate the comments of the anonymous referees for stimulating us to clarify the presentation of the Methods section. This work was supported in part by NIH Grant (S.S.) RO1 GM46535 (subcontract from Texas A & M), NSF Grant (F.S.) ASC 92 09502 RIA, DOE Grant (M.H.) DE-FG02-91ER25099, and a grant from FMC Corporation (S.S.). We thank the National Center for Supercomputing Applications for providing access to various computational resources.

REFERENCES

1. Debye, P., Hückel, E. On the theory of electrolytes: Freezing point depression and related phenomena. In: "The Collected Works of Peter J.W. Debye," New York: Interscience, 1923: pp. 217–263.
2. Warshel, A., Russell, S. T. Calculations of electrostatic interactions in biological systems and in solution. Quart. Rev. Biophys. 17:238–422, 1984.
3. Harvey, S. C. Treatment of electrostatic effects in macromolecular modeling. Proteins 5:78–92, 1989.
4. Davis, M. E., McCammon, J. A. Electrostatics in biomolecular structure and dynamics. Chem. Rev. 90:509–521, 1990.
5. Sharp, K. A., Honig, B. Electrostatic interactions in macromolecules: Theory and applications. Annu. Rev. Biophys. Biophys. Chem. 19:301–332, 1990.
6. Davis, M. E., Madura, J. D., Luty, B. A., McCammon, J. A. Electrostatics and diffusion of molecules in solution: Simulations with the University of Houston Brownian dynamics program. Comp. Phys. Commun. 62:187–197, 1991.
7. Niedermeier, C., Schulten, K. Molecular dynamics simulations in heterogeneous dielectrica and Debye-Hückel media—application to the protein bovine pancreatic trypsin inhibitor. Technical report, Department of Physics and Beckman Institute, University of Illinois at Urbana-Champaign, 1990.
8. Sharp, K. A. Incorporating solvent and ion screening into molecular dynamics using the finite-difference Poisson-Boltzmann method. J. Comput. Chem. 12:454–468, 1991.
9. Holst, M. Multilevel methods for the Poisson-Boltzmann equation. Ph.D. thesis, Numerical Computing Group, Department of Computer Science, University of Illinois at Urbana-Champaign, 1993.
10. Tanford, C. "Physical Chemistry of Macromolecules." New York: Wiley, 1961.
11. Brenner, S. L., Roberts, R. E. A variational solution of the Poisson-Boltzmann equation for a spherical colloidal particle. J. Phys. Chem. 77:2367–2370, 1973.
12. Fujita, H. On the nonlinear equations $\Delta u + e^u = 0$ and $\partial v/\partial t = \Delta v + e^v$. Bull. Am. Math. Soc. 75:132–135, 1969.
13. Ting, A. C., Chen, H. H., Lee, Y. C. Exact solutions of a nonlinear boundary value problem: the vortices of the two-dimensional sinh-Poisson equation. Physica D 26:37–66, 1987.
14. Davis, M. E., McCammon, J. A. Solving the finite difference linearized Poisson-Boltzmann equation: A comparison of relaxation and conjugate gradient methods. J. Comput. Chem. 10(3):386–391, 1989.
15. Gilson, M. K., Sharp, K. A., Honig, B. H. Calculating the electrostatic potential of molecules in solution: Method and error assessment. J. Comput. Chem. 9:327–335, 1988.
16. Juffer, A. H., Botta, E. F. F., van Keulen, B. A. M., van der Ploeg, A., Berendsen, H. J. C. The electric potential of a macromolecule in a solvent: A fundamental approach. J. Comput. Phys. 97:144–171, 1991.
17. Nicholls, A., Honig, B. A rapid finite difference algorithm, utilizing successive over-relaxation to solve the Poisson-Boltzmann equation. J. Comput. Chem. 12:435–445, 1991.
18. Yoon, B. J., Lenhoff, A. M. A boundary element method for molecular electrostatics with electrolyte effects. J. Comput. Chem. 11:1080–1086, 1990.
19. Warwicker, J., Watson, H. C. Calculation of the electric potential in the active site cleft due to α-helix dipoles. J. Mol. Biol. 157:671–679, 1982.
20. Ortega, J. M., Rheinboldt, W. C. "Iterative Solution of Nonlinear Equations in Several Variables." New York: Academic Press, 1970.
21. Allison, S. A., Sines, J. J., Wierzbicki, A. Solutions of the full Poisson-Boltzmann equation with application to diffusion-controlled reactions. J. Phys. Chem. 93:5819–5823, 1989.
22. Jayaram, B., Sharp, K. A., Honig, B. The electrostatic potential of B-DNA. Biopolymers 28:975–993, 1989.
23. Luty, B. A., Davis, M. E., McCammon, J. A. Solving the finite-difference non-linear Poisson-Boltzmann equation. J. Comput. Chem. 13:1114–1118, 1992.
24. Rashin, A. A., Malinsky, J. New method for the computation of ionic distribution around rod-like polyelectrolytes with helical distribution of charges. I. General approach and a nonlinearized Poisson-Boltzmann equation. J. Comput. Chem. 12:981–993, 1991.
25. Sharp, K. A., Honig, B. Calculating total electrostatic energies with the nonlinear Poisson-Boltzmann equation. J. Phys. Chem. 94:7684–7692, 1990.
26. Oberoi, H., Allewell, N. M. Multigrid solution of the nonlinear Poisson-Boltzmann equation and calculation of titration curves. Biophys. J. 65:48–55, 1993.
27. Fletcher, R., and Reeves, C. Function minimization by conjugate gradients. Comput. J. 7:149–154, 1964.
28. Holst, M., Saied, F. Multigrid solution of the Poisson-Boltzmann equation. J. Comput. Chem. 14:105–113, 1993.
29. Bank, R. E., Dupont, T. F. An optimal order process for solving finite element equations. Math. Comp. 36:35–51, 1981.
30. Bramble, J. H., Pasciak, J. E. New convergence estimates for multigrid algorithms. Math. Comp. 49:311–329, 1987.
31. Hackbusch, W. "Multi-grid Methods and Applications." Springer-Verlag, Berlin, Germany, 1985.
32. de Zeeuw, P. M. Nonlinear multigrid applied to a one-dimensional stationary semiconductor model. SIAM J. Sci. Statist. Comput. 13:512–530, 1992.
33. Hackbusch, W. On the fast solutions of nonlinear elliptic equations. Numer. Math. 32:83–95, 1979.
34. Hackbusch, W., Reusken, A. On global multigrid convergence for nonlinear problems. In: "Robust Multigrid Methods." Hackbusch, W. ed. Vieweg: Braunschweig, 1988, pp. 105–113.
35. Hackbusch, W., Reusken, A. Analysis of a damped nonlinear multilevel method. Numer. Math. 55:225–246, 1989.
36. Reusken, A. Convergence of the multigrid full approximation scheme for a class of elliptic mildly nonlinear boundary value problems. Numer. Math. 52:251–277, 1988.
37. Reusken, A. Convergence of the multilevel full approximation scheme including the V-cycle. Numer. Math. 53: 663–686, 1988.
38. Brandt, A. Multi-level adaptive solutions to boundary-value problems. Math. Comp. 31:333–390, 1977.
39. Briggs, W. L. "A Multigrid Tutorial." Philadelphia, PA: SIAM, 1987.
40. Klapper, I., Hagstrom, R., Fine, R., Sharp, K., Honig, B. Focusing of electric fields in the active site of cu-zn superoxide dismutase: Effects of ionic strength and amino-acid modification. Proteins 1:47–59, 1986.
41. Varga, R. S. "Matrix Iterative Analysis." Englewood Cliffs, NJ: Prentice-Hall, 1962.

42. Davis, H. T. "Introduction to Nonlinear Differential and Integral Equations." New York: Dover, 1960.
43. Bank, R. E., Rose, D. J. Analysis of a multilevel iterative method for nonlinear finite element equations. Math. Comp. 39:453–465, 1982.
44. Bank, R. E. A multi-level iterative method for nonlinear elliptic equation. In: "Proceedings of the Conference on Elliptic Problem Solvers." Birkhoff, G., Schoenstadt, A., eds. New York: Academic Press, 1984.
45. Bank, R. E., Rose, D. J. Parameter selection for Newton-like methods applicable to nonlinear partial differential equations. SIAM J. Numer. Anal. 17:806–822, 1980.
46. Bank, R. E., Rose, D. J. Global approximate Newton methods. Numer. Math. 37:279–295, 1981.
47. Dembo, R. S., Eisenstat, S. C., Steihaug, T. Inexact Newton methods. SIAM J. Numer. Anal. 19:400–408, 1982.
48. Eisenstat, S. C., Walker, H. F. Globally convergent inexact Newton methods. Technical report, Dept. of Mathematics and Statistics, Utah State University, 1992.
49. Holst, M., Saied, F. Multilevel methods for three-dimensional nonlinear Poisson equations. Technical Report no. UIUCDCS-R-93-1821, Numerical Computing Group, Department of Computer Science, University of Illinois at Urbana-Champaign, 1993.
50. Jorgensen, W. L., Tirado-Rives, J. The OPLS [optimized potentials for liquid simulations] potential functions for crystals of cyclic peptides and crambin. J. Am. Chem. Soc. 110:1657–1669, 1988.
51. Teeter, M. M. Water structure of a hydrophobic protein at atomic resolution: Pentagon rings of water molecules in crystals of crambin. Proc. Natl. Acad. Sci. U.S.A. 81:6014–6018, 1984.
52. Wlodawer, A., Nachman, J., Gilliland, C. L., Gallagher, W., Woodward, C. Structure of form III crystals of bovine pancreatic trypsin inhibitor. J. Mol. Biol. 198:469–480, 1987.
53. Sheriff, S., Silverton, S., Padlan, E. A., Cohen, G. H., Smith-Gill, S. J., Finzel, B. C., Davies, D. R. Three-dimensional structure of antibody-antigen complex. Proc. Natl. Acad. Sci. U.S.A. 84:8075–8079, 1987.
54. Tainer, J. A., Getzoff, E. D., Beem, K. M., Richardson, J. S., Richardson, D. C. Determination and analysis of the 2 Å structure of copper, zinc superoxide dismutase. J. Mol. Biol. 160:181–216, 1982.
55. Tainer, J. A., Getzoff, E. D., Richardson, J. S., Richardson, D. C. Structure and mechanism of copper, zinc superoxide dismutase. Nature (London) 306:284–287, 1983.
56. Arnold, E., Vriend, G., Luo, M., Griffith, J. P., Kamer, G., Erickson, J. W., Johnson, J. E., Rossmann, M. G. The structure determination of a common cold virus, human rhinovirus 14. Acta Crystallogr. Sect. A 43:346–361, 1987.
57. Hyde, C. C., Ahmed, S. A., Padlan, E. A., Miles, E. W., Davies, D. R. Three-dimensional structure of the tryptophan synthase $\alpha_2\beta_2$ multienzyme complex from *Salmonella typhimurium*. J. Biol. Chem. 263:17857–17871, 1988.
58. Kozack, R. E., Subramaniam, S. Brownian dynamics simulations of molecular recognition in an antibody-antigen system. Protein Sci. 2:915–926, 1993.
59. Holst, M., Kozack, R. E., Saied, F., Subramaniam, S. Protein electrostatics: Rapid multilvel solution of the full nonlinear poisson-boltzmann equation. Submitted.
60. Davis, M. E., Madura, J. D., Sines, J., Luty, B. A., Allison, S. A., McCammon, J. A. Diffusion-controlled enzymatic reactions. Methods Enzymol. 202:478–497, 1990.
61. Ermak, D. L., McCammon, J. A. Brownian dynamics with hydrodynamic interactions. J. Chem. Phys. 69:1352–1360, 1978.
62. Northrup, S. H., Smith, J. D., Boles, J. O., Reynolds, J. C. L. The effect of dipole moment on diffusion controlled biomolecular reaction rates. J. Chem. Phys. 84:5536–5544, 1986.
63. Northrup, S. H., Erickson, H. P. Kinetics of protein-protein association explained by brownian dynamics computer simulation. Proc. Natl. Acad. Sci. U.S.A. 89:3338–3342, 1992.
64. Raman, C. S., Jemmerson, R., Nall, B. T., Allen, M. J. Diffusion-limited rates for monoclonal antibody binding to cytochrome *c*. Biochemistry 31:10370–10379, 1992.
65. Smith-Gill, S. J., Wilson, A. C., Potter, M., Prager, E. M., Feldmann, R. J., Mainhart, C. R. Mapping the antigenic epitope for a monoclonal antibody against lysozyme. J. Immunol. 128:314–322, 1982.
66. Lavoie, T. B., Kam-Morgan, L. N. W., Hartman, A. B., Mallett, C. P., Sheriff, S., Saroff, D. G., Mainhart, C. R. Hamel, P. A., Kirsch, J. F., Wilson, A. C., Smith-Gill, S. J. Structure-function relationships in high-affinity antibodies to lysozyme. In: "The Immune Response to Structurally Defined Proteins: The Lysozyme Model." Smith-Gill, S.J., Secarz, E.E., eds. New York: Adenine Press, 1989.
67. Allison, S. A., Ganti, G., McCammon, J. A. Simulation of the diffusion-controlled reaction between superoxide and superoxide dismutase i. simple models. Biopolymers 24: 1323–1336, 1985.
68. Allison, S. A., Northrup, S. H., McCammon, J. A. Simulation of biomolecular diffusion and complex formation. Biophys. J. 49:167–175, 1986.
69. Sharp, K., Fine, R., Honig, B. Computer simulation of the diffusion of a substrate to the active site of an enzyme. Nature (London) 236:1460–1463, 1987.
70. Sharp, K., Fine, R., Schulten, K., Honig, B. Brownian dynamics simulation of diffusion to irregular bodies. J. Phys. Chem. 91:3624–3631, 1987.
71. Allison, S. A., Bacquet, R. J., McCammon, J. A. Simulation of the diffusion-controlled reaction between superoxide and superoxide dismutase II. Detailed models. Biopolymers 27:251–269, 1988.
72. Sines, J. J., Allison, S. A., McCammon, J. A. Point-charge distributions and electrostatic steering in enzyme-substrate encounter: Brownian dynamics of modified copper-zinc superoxide dismutases. Biochemistry 29:9403–9412, 1990.
73. Sines, J. J., McCammon, J. A., Allison, S. A. Kinetic effects of multiple charge modifications in enzyme-substrate reactions: Brownian dynamics simulations of cu,zn superoxide dismutases. J. Comput. Chem. 13:66–69, 1992.
74. Getzoff, E. D., Fisher, C. L., Parge, H. E., Viezzoli, M. S., Banci, L., Hallewell, R. A. Faster superoxide dismutase mutants designed by enhancing electrostatic steering. Nature (London) 330:84–86, 1992.
75. Shen, J., Subramaniam, S., Wong, C. F., McCammon, J. A. Superoxide dismutase: Fluctuations in the structure and solvation of the active site channel studied by molecular dynamics simulation. Biopolymers 28:2085–2096, 1989.
76. Hyde, C. C., Miles, E. W. The tryptophan synthase multienzyme complex: Exploring structure-function relationships with x-ray crystallography and mutagenesis. Bio/Technology 8:27–32, 1990.