Fast Boundary Element Method for the Linear Poisson−Boltzmann Equation

Alexander H. Boschitsch,*,[†] Marcia O. Fenley,*,[‡,‖] and Huan-Xiang Zhou[§,⊥]

Continuum Dynamics, Inc., 34 Lexington Avenue, Ewing, New Jersey 08618-2302, Department of Physics, Washington University, St. Louis, Missouri 63130-4899, and Department of Physics, Drexel University, Philadelphia, Pennsylvania 19104

Received: September 24, 2001; In Final Form: December 13, 2001

This article summarizes the development of a fast boundary element method for the linear Poisson−Boltzmann equation governing biomolecular electrostatics. Unlike previous fast boundary element implementations, the present treatment accommodates finite salt concentrations thus enabling the study of biomolecular electrostatics under realistic physiological conditions. This is achieved by using multipole expansions specifically designed for the exponentially decaying Green's function of the linear Poisson−Boltzmann equation. The particular formulation adopted in the boundary element treatment directly affects the numerical conditioning and thus convergence behavior of the method. Therefore, the formulation and reasons for its choice are first presented. Next, the multipole approximation and its use in the context of a fast boundary element method are described together with the iteration method employed to extract the surface distributions. The method is then subjected to a series of computational tests involving a sphere with interior charges. The purpose of these tests is to assess accuracy and verify the anticipated computational performance trends. Finally, the salt dependence of electrostatic properties of several biomolecular systems (alanine dipeptide, barnase, barstar, and coiled coil tetramer) is examined with the method and the results are compared with finite difference Poisson−Boltzmann codes.

I. Introduction

The properties and function of numerous charged biomolecules and their complexes with other molecules are dependent on the ionic strength of the environment. For example, the conformational stability of highly charged peptides and proteins is greatly affected by changes in salt concentration. Similarly, the binding affinities and association rates of biomolecular complexes are also strongly dependent on salt concentration. Therefore, computational tools that reliably and accurately predict ionic strength dependent electrostatic interactions are essential for an improved understanding of many biological processes. Moreover, to offer the user the ability to address large molecules using readily accessible computers, the CPU and storage demands imposed by such tools must be kept at a minimum.

The different theoretical approaches used to model salt effects in biomolecular systems can be divided into two broad categories according to whether they employ an explicit or implicit solvent model. Explicit solvent models adopt microscopic representations of both solute (e.g., biomolecule) and solvent molecules. Typically, explicit solvent-based approaches employ potential energy functions and sample the conformational space by either molecular dynamics or Monte Carlo techniques. Explicit solvent approaches produce accurate results, but are very computer intensive. Molecular dynamics of biomolecules immersed in salt solutions entail considerable computational effort since they involve a large number of ions and water molecules and require accurate ion−water, ion−ion, and ion−solute potential functions. Obtaining thermodynamic quantities, such as solvation and binding free energies, from free energy simulations of biomolecules immersed in aqueous salt solutions is even more challenging. The introduction of efficient particle-mesh Ewald (PME) algorithms[1,2] which accurately account for long-range Coulombic interactions has promoted the popularity of biomolecular dynamics simulations and allowed larger systems to be considered. However, despite the advent of such tools and advances in computer power, molecular dynamics simulations of ionic strength effects in biomolecular systems are still not practical using an explicit solvent model.[3]

Implicit solvent models adopt a semi-microscopic treatment of the solute, but characterize the solvent in terms of its macroscopic physical properties (e.g., dielectric constant, ionic strength). This allows ionic strength effects to be accurately reproduced at much lower computational cost than with explicit solvent approaches. For this reason, implicit solvent models such as those based on the Poisson−Boltzmann equation (PBE) are now widely used. In particular, the linear Poisson−Boltzmann approach considered here has been successfully used to account for the salt dependence of a variety of thermodynamic quantities such as binding free energies,[4] pK shifts,[5−7] and biomolecular association rate constants.[8,9] The Poisson−Boltzmann equation has also been coupled to quantum chemistry methods (e.g., ref 10) and used in conjunction with molecular dynamics to obtain relative free energies (e.g., binding free energies), which include both molecular mechanical free energies and solvation free energies.[11]

Analytical solutions to the linear Poisson−Boltzmann equation are only available for a limited number of cases involving

* Correspondence should be addressed to A.H.B: telephone: (609) 538-0444, fax: 538-0464, e-mail: alex@continuum-dynamics.com; and MOF: telephone: (850) 644-7961, fax: (850) 644-7244, e-mail: mfenley@sb.fsu.edu.

[†] Continuum Dynamics, Inc.

[‡] Washington University.

[§] Drexel University.

[‖] Current address: Institute of Molecular Biophysics, Florida State University, Tallahassee, FL 32306.

[⊥] Address after July 1, 2001: Institute of Molecular Biophysics and Department of Physics, Florida State University, Tallahassee, FL 32306.

10.1021/jp013607q CCC: $22.00
Published on Web 02/06/2002

idealized geometries such as spheres and cylinders. For the complex molecular surfaces and arbitrary charge distribution of real molecules, the linear Poisson−Boltzmann equation must be solved numerically using one of various numerical techniques. The most popular means of solving the linear Poisson−Boltzmann equation is the finite difference method (FDM).[12−14] In this approach, a discrete approximation to the governing partial differential equations is procured upon a volume-filling grid. Ideally, a boundary conforming grid (i.e., one that does not intersect the molecular surface) is preferred,[15] but such a grid is difficult to generate about a complex molecular shape. Therefore, in most implementations, such as the widely used UHBD and DelPhi codes,[13,16−18] a regular lattice is laid over the molecule and cells are allowed to straddle the molecular surface. A regular lattice arrangement also lends itself well to efficient multigrid solution techniques for solving the algebraic equations resulting from the discretization of the partial differential equations.[19] The finite element method differs from the finite difference approach in that variational principles rather than finite difference approximations are used to derive the discrete equations.[15,20−22] A major reason for adopting a finite element approach is that it accommodates unstructured grids which offer improved geometric flexibility and variable mesh spacing, albeit at considerably higher per-node storage and CPU costs compared to a regular lattice arrangement.

Solutions to the linear PBE can also be expressed in terms of singular surface integrals. This representation is the basis of the class of numerical procedures known as the boundary element method (BEM). In the basic approach, the molecular surface is approximated by a collection of boundary elements whose electrostatic potential and normal potential gradients are to be determined. Once these surface values are known, the electrostatic potential at all other locations can be computed explicitly using the appropriate boundary integrals. Equations for the surface values are derived by considering the limiting forms of the electrostatic potentials as one approaches the molecular surface from the interior and exterior regions. Numerous boundary element codes exist for solving the Poisson and Poisson−Boltzmann equations (e.g., refs 23−31). Because of the basic differences in formulation, the BEM enjoys certain intrinsic advantages over the FDM including the following:

(i) The solution is completely characterized in terms of surface distributions. This "lowering of dimensionality" results in fewer unknowns compared to the FDM where the solution is developed over the entire three-dimensional volume.

(ii) Far-field boundary conditions are enforced exactly—in the FDM the grid is of finite extent so that the solution at the outer boundaries must be approximated or calculated by alternate means.

(iii) The surface geometry is represented to high precision since the orientation and position of each boundary element are close to those of the true surface. In the FDM only an approximate "staircase" rendering of the surface is possible since it is straddled by the grid cells. Hence, for comparable characteristic grid spacings, the discrete surface approximation adopted in the BEM will generally deviate much less from the true surface than in the FDM.

(iv) The exact surface boundary conditions are explicitly enforced. In the FDM, continuity in the electrostatic potential is imposed, but the constraint upon the normal electrostatic field is only weakly enforced in an average sense. Unless a boundary conforming mesh is employed, there is no assurance that the normal field at the surface converges to the correct value as the grid spacing is reduced.

(v) The electrostatic field of point charges and surface elements is reproduced exactly at all points in space using the explicit mathematical expressions. This feature, in conjunction with the inherent error averaging properties of an integral method, confers high accuracy to the BEM.

In light of these advantages, the BEM, especially when accelerated with the fast methods described below, can provide a computational approach superior to FDM. However, the BEM is also valuable as an independent analysis tool for confirming results. While the FDM and BEM use very different mathematical and numerical approaches, when applied to the same governing equation (as done here) they should converge to identical answers as mesh resolution is increased. Therefore, applying FDM and BEM to the same problem provides a way of establishing high confidence in the predictions (i.e., the FDM and BEM results are in close agreement) or uncovering modeling problems (i.e., they show significant differences) that indicate the need for reexamination or higher resolution. The need for independent analyses of this kind has grown as increasingly complex biomolecular systems come under examination and converged results are harder to establish.

In the conventional BEM, these advantages are offset by the high computational cost incurred in evaluating the mutual interactions between all elements. For a problem involving N elements, the calculation of the potential at all elements entails $O(N^2)$ interactions. Moreover, each interaction involves singular integrals whose analytical and/or numerical evaluation is expensive. Inversion of the matrix to solve for the surface potentials requires $O(N^3)$ operations for direct inversion and $O(mN^2)$ for m steps of an iterative inversion method. Therefore, in practice, application of boundary element based Poisson or linear PBE algorithms has been limited to small numbers of boundary elements (up to $O(10^3)$) and consequently small to medium-sized molecules.

To address these costs and accommodate large biomolecular systems, fast multipole algorithms (FMA) have been employed.[32] These methods carefully combine a hierarchical grouping procedure that assembles the elements into nested groups together with a low-cost multipole-based approximation method for evaluating the potential induced by an element group, to achieve substantial reductions in computational requirements. For example, the use of fast multipole algorithms in BEM-based Poisson equation solvers has led to orders of magnitude reductions in both storage and CPU compared to conventional implementations.[33,34] However, since the Poisson equation addresses only zero salt conditions, such fast BEM Poisson programs are limited. To our knowledge, fast multipole methods have not been previously incorporated into a BEM for solving the Poisson−Boltzmann equation. Here, the fast BEM is extended to accommodate both the Poisson and linear Poisson−Boltzmann equations at finite salt concentrations. This is accomplished by incorporating new multipole expansions that are explicitly designed for the exponential Green's function associated with the PBE.[35] In the limit of zero salt concentration, these expansions revert to the classical expansions for Coulombic potentials.

Following a review of the boundary element method applied to the linear Poisson−Boltzmann equation, this article describes how to formulate and implement a fast multipole algorithm to reduce computational costs. This section also introduces a mesh adaptation method that allows high accuracy to be achieved with fewer elements and provides a useful analytical result for validating any linear Poisson−Boltzmann solver. This section is followed by an assessment of the computational performance and accuracy of the method when applied to the idealized case of a spherical cavity containing interior charges for which analytical solutions are known. Finally, the method is used to calculate the electrostatic properties of realistic biomolecules including different conformers of alanine dipeptide, barnase and barstar, survivin, coiled coil tetramer, and FinO.

II. Methodology

A. Formulation. In this work, the solute molecule is treated as a region, Ω_1, of low dielectric constant (ϵ_1) containing the atom centered point charges and the surrounding solvent region, Ω_2, is modeled as a structureless dielectric continuum having dielectric constant ϵ_2. The mobile ions are treated implicitly and obey the Boltzmann distribution. Also, the Stern layer is omitted. The solute is described in terms of its three-dimensional structure obtained from NMR/X-ray crystallography or molecular modeling techniques. The molecular surface, S, separating the interior and exterior regions, Ω_1 and Ω_2, can be taken as either the van der Waals, solvent-accessible, or solvent-excluded surfaces.[36] The choice of which molecular surface to use as the dielectric interface is still open to debate.[37] For N_q fixed interior point charges, Q_k, located at atomic centers, ρ_k, the electrostatic potential, $\phi(R)$, at a point, R, in the interior region, Ω_1, is governed by the Poisson equation:

$$\nabla^2\phi(\vec{R}) = -\sum_{k=1}^{N_q} q_k \delta(\vec{R} - \vec{\rho}_k) \tag{1}$$

where $\delta(\bullet)$ is the Dirac delta function and $q_k = Q_k/\epsilon_1$. In the exterior region, Ω_2, the electrostatic potential, μ, obeys the linear Poisson−Boltzmann equation:

$$(\nabla^2 - \kappa^2)\mu(\vec{R}) = 0 \tag{2}$$

The Debye−Hückel screening parameter, κ, representing the attenuation of electrostatic interactions by the presence of salt in the solution, is related to the ionic strength of the aqueous salt solution, I, according to

$$\kappa^2 = \frac{8\pi e^2 I}{\epsilon_2 k_B T} \tag{3}$$

where e is the elementary charge, k_B is the Boltzmann constant, and T is the solution temperature.

At the common interface between the two regions (i.e., dielectric boundary), the electrostatic potential and the normal gradient of the electric displacement are continuous; thus

$$\phi(\vec{R}) = \mu(\vec{R}), \frac{\partial\phi}{\partial n}(\vec{R}) = \epsilon\frac{\partial\mu}{\partial n}(\vec{R}); \vec{R} \in \mathrm{S} \tag{4a,b}$$

where $\epsilon = \epsilon_2/\epsilon_1$ and $\vec{n}$ is the surface normal directed outward from region Ω_1 to Ω_2. The regularity conditions that both $R\mu(\vec{R})$ and $R^2\nabla\mu(\vec{R})$ remain finite as $R = |\vec{R}| \rightarrow \infty$ also apply.

The general solution for the electrostatic potential in each region is given by[23]

$$\phi(\vec{R}) = \oint_S \left[G_0(\vec{R},\vec{\rho})\frac{\partial\phi}{\partial n}(\vec{\rho}) - \frac{\partial G_0}{\partial n}(\vec{R},\vec{\rho})\,\phi(\vec{\rho})\right] \mathrm{d}S + \sum_{k=1}^{N_q} q_k G_0(\vec{R},\vec{\rho}_k), \vec{R} \in \Omega_1 \tag{5}$$

$$\mu(\vec{R}) = \oint_S \left[-G_\kappa(\vec{R},\vec{\rho})\frac{\partial\mu}{\partial n}(\vec{\rho}) + \frac{\partial G_\kappa}{\partial n}(\vec{R},\vec{\rho}\ )\,\mu(\vec{\rho})\right] \mathrm{d}S, \vec{R} \in \Omega_2 \tag{6}$$

where the fundamental solution for the linear Poisson−Boltzmann equation

$$G_\kappa(\vec{R},\vec{\rho}) = \frac{\exp[-\kappa|\vec{R} - \vec{\rho}|]}{4\pi|\vec{R} - \vec{\rho}|} \tag{7}$$

Note that $G_0(\vec{R},\vec{\rho})$ is the fundamental solution for the Poisson equation. Using well-known limiting relations, (5) and (6) reduce to the following forms as the evaluation point, $\vec{R}$, approaches the surface, S:

$$\lim_{\vec{R}\rightarrow S-} \phi(\vec{R}) = \frac{1}{2}\phi(\vec{R}) + \oint_S \left[G_0(\vec{R},\vec{\rho})\frac{\partial\phi}{\partial n}(\vec{\rho}) - \frac{\partial G_0}{\partial n}(\vec{R},\vec{\rho})\,\phi(\rho)\right] \mathrm{d}S + \sum_{k=1}^{N_q} q_k G_0(\vec{R},\vec{\rho}_k) \tag{8}$$

$$\lim_{\vec{R}\rightarrow S^+} \mu(\vec{R}) = \frac{1}{2}\mu(\vec{R}) + \oint_S \left[-G_\kappa(\vec{R},\vec{\rho})\frac{\partial\mu}{\partial n}(\vec{\rho}) + \frac{\partial G_\kappa}{\partial n}(\vec{R},\vec{\rho})\,\mu(\vec{\rho})\right] \mathrm{d}S \tag{9}$$

where a locally smooth surface is assumed and S^- and S^+ represent the surfaces located an infinitesimal amount below (i.e., in Ω_1) and above (in Ω_2) S, respectively.

Equations 8 and 9 in conjunction with the specified boundary conditions, (4), comprise a complete set of equations that can be solved for the surface distributions of ϕ and $\partial\phi/\partial n$. A so-called "nonderivative" boundary element (BE) formulation derived directly from these equations is adopted by several groups.[29,31,38] Unfortunately, the resulting system of linear algebraic equations becomes increasingly ill-conditioned as the number of boundary elements grows. For this reason, the alternate "derivative" BE formulation introduced by Juffer and co-workers[23] is used which maintains a well-conditioned system of algebraic equations for arbitrary boundary element counts.[24] Under this approach, (8) and (9) are linearly combined to obtain

$$\frac{1}{2}(1 + \epsilon)\phi(\vec{R}) = \oint_S \left[(G_0 - G_\kappa)\frac{\partial\phi}{\partial n}(\vec{\rho}) - \left(\frac{\partial G_0}{\partial n} - \epsilon\frac{\partial G_\kappa}{\partial n}\right)\phi(\vec{\rho})\right] \mathrm{d}S + \sum_{k=1}^{N_q} q_k G_0(\vec{R},\vec{\rho}_k) \tag{10}$$

Note that (4) has been used to eliminate μ and $\partial\mu/\partial n$. A second equation is obtained by differentiating (8) and (9) with respect to the normal, $\vec{n}_0 = \vec{n}(\vec{R})$, and combining the results to produce

$$\frac{1}{2}\left(1 + \left(\frac{1}{\epsilon}\right)\right)\frac{\partial\phi}{\partial n_0}(\vec{R}) = \oint_S \left[\left(\frac{\partial G_0}{\partial n_0} - \frac{1}{\epsilon}\frac{\partial G_\kappa}{\partial n_0}\right)\frac{\partial\phi}{\partial n}(\vec{\rho}) - \left(\frac{\partial^2 G_0}{\partial n_0\,\partial n} - \frac{\partial^2 G_\kappa}{\partial n_0\,\partial n}\right)\phi(\vec{\rho})\right] \mathrm{d}S + \sum_{k=1}^{N_q} q_k \frac{\partial G_0}{\partial n_0}(\vec{R},\vec{\rho}_k) \tag{11}$$

The linear combination (11) is unique in that nonintegrable singularities are completely eliminated. In any other combination, the second-order gradients give rise to $O(1/r^3)$ hypersingular integrals.[23]

B. Discretization and Quadrature. In the present work, a discrete approximation to the integral equations, (10) and (11), is obtained by dividing the molecular surface into a collection of boundary elements (triangles and quadrilaterals) and approximating the surface solutions, ϕ and $\partial\phi/\partial n$, by piecewise constant distributions. This combination constitutes a "low-order" model since the higher order effects of surface curvature and solution variations over each element are neglected. Curvilinear treatments that capture these higher order effects generally deliver better accuracy, thus allowing fewer elements to be used, but require more CPU per element and more elaborate quadrature schemes. Conversely, low-order descriptions[26,29,34] facilitate the construction of robust surface quadrature rules for the singular integrands, are less sensitive to surface mesh imperfections such as gaps or overlaps, and are ideally suited to mesh adaptation. Note, however, that the fast multipole algorithm can be equally well applied to both low- and high-order methods.

Let the molecular surface be represented by a collection of N boundary elements, j, each having area, normal, and centroid denoted by A_j, n_j, and $\vec{R}_j$, respectively. Under this representation equations, (10) and (11), governing the electrostatic potential $\phi_i = \phi(\vec{R}_i)$ and its normal potential gradient $h_i = \partial\phi(R_i)/\partial n_i$, at the center of element, i, reduce to

$$\left(\frac{1}{2}\right)(1+\epsilon)\phi_i = \sum_{\substack{j=1\\ j\neq 1}}^{N} A_{1,ij}\phi_j + \sum_{j=1}^{N} A_{2,ij}h_j + \sum_{k=1}^{N_q} q_k G_0(\vec{R}_i,\vec{\rho}_k) \equiv r_{1i} \quad (12a)$$

$$\frac{1}{2}\left(1+\frac{1}{\epsilon}\right)h_i = \sum_{j=1}^{N} A_{3,ij}\phi_j + \sum_{\substack{j=1\\ j\neq 1}}^{N} A_{4,ij}h_j + \sum_{k=1}^{N_q} q_\mathrm{k}\frac{\partial G_0}{\partial n_0}(\vec{R}_i,\vec{\rho}_k) \equiv r_{2i} \quad (12b)$$

where

$$A_{1,ij} = \int_{A_j}\frac{\partial}{\partial n}(\epsilon G_\kappa - G_0)\,\mathrm{d}A \quad (13a)$$

$$A_{2,ij} = \int_{A_j} G_0 - G_\kappa\,\mathrm{d}A \quad (13b)$$

$$A_{3,ij} = \int_{A_j}\left(\frac{\partial^2 G_\kappa}{\partial n_i\,\partial n} - \frac{\partial^2 G_0}{\partial n_i\,\partial n}\right)\mathrm{d}A \quad (13c)$$

$$A_{4,ij} = \int_{A_j}\frac{\partial}{\partial n_i}\left(G_0 - \frac{1}{\epsilon}G_\kappa\right)\mathrm{d}A \quad (13d)$$

Note that for planar surface elements the diagonal entries of these coefficients reduce to $A_{1,ii} = A_{4,ii} = 0$, where the Cauchy principal values of the quadratures are *not* included since they are already accounted for on the left-hand side of (12). The surface solutions, ϕ_i and h_i, are obtained by inverting the $2N$ equations, (12). Once these surface values are known, the electrostatic potential can be computed in the interior and exterior regions using the discrete forms of (5) and (6).

The accurate and efficient evaluation of the boundary element integrals is essential to obtaining reliable predictions of the electrostatic potentials. However, the procedure is only briefly described here since the emphasis is upon the fast summation and inversion techniques. The immediate difficulty concerns the $1/r^p$ singularities present in the integrands. Gauss quadrature methods developed for smooth functions are ineffective near such singularities and can require hundreds of quadrature points to achieve even moderate accuracy. An improved approach is to transform to polar coordinates which essentially reduces the order of the singularity by 1 and thus is useful for integrands of $O(1/r)$. However, additional and rather elaborate procedures must be employed to tackle the more complicated singularities associated with the normal derivatives, $\partial G_\kappa/\partial n$. In the current work, advantage is taken of the fact that for zero ion screening (pure Coulombic potentials) and planar elements the quadratures can be evaluated exactly.[39−41] For finite salt concentration ($\kappa \neq 0$) such analytical expressions do not seem available. Nevertheless, it is possible to separate each integral into (i) a singular component corresponding to $\kappa = 0$ which is evaluated exactly and (ii) a smooth, nonsingular correction term that accounts for the finite salt effects and can be accurately evaluated by Gauss quadrature. The regular correction term vanishes as $\kappa \to 0$ so that in the absence of ion screening all integrals are performed analytically.

To illustrate, consider the expression for $A_{1,ij}$. Starting with

$$\frac{\partial G_\kappa}{\partial n} = \vec{n}\cdot\nabla G_\kappa = \vec{n}\cdot\left[\frac{\mathrm{e}^{-\kappa r}(1+\kappa r)}{r^3}(\vec{R}-\vec{\rho})\right],\ r = |\vec{R}-\vec{\rho}| \quad (14a,b)$$

and replacing the exponential factor, $\exp\{-\kappa r\}$, by its Taylor series, then one can readily identify all singular terms. This leads to

$$\nabla G_\kappa = -\frac{\mathrm{e}^{-\kappa r}(1+\kappa r)}{r^3}(\vec{R}-\vec{\rho}) = -\frac{1}{r^3}(\vec{R}-\vec{\rho}) - d_1(\kappa,r)\frac{(\vec{R}-\vec{\rho})}{r} \quad (15)$$

where the nonsingular function

$$d_1(\kappa,r) = \frac{\mathrm{e}^{-\kappa r}(1+\kappa r) - 1}{4\pi r^2} \quad (16)$$

It is easy to show that $4\pi d_1(\kappa,r) \to -(\kappa^2/2)$ as $r \to 0$; also, $d_1(\kappa,r) \to 0$ as $\kappa \to 0$ so that this term vanishes with decreasing salt concentration. For small r, $d_1(\kappa,r)$ is accurately and robustly evaluated in terms of its Taylor series expansion; for larger values of r, (16) is used directly. Substituting (15) into (14) and then into (13a) leads to

$$A_{1,ij} = \int_{A_j}\frac{\partial}{\partial n}(\epsilon G_\kappa - G_0)\,\mathrm{d}A = (\epsilon - 1)\int_{A_j}\frac{\partial G_0}{\partial n_j}\,\mathrm{d}A + \int_{A_j}\epsilon d_1(\kappa,r)\frac{\vec{n}_j\cdot(\vec{R}-\vec{\rho})}{r}\,\mathrm{d}A \quad (17)$$

The first integral on the right-hand side involves only the Coulombic potential and is evaluated in closed form.[39−41] The second integral involves the smooth function, $d_1(\kappa,r)$, and the nonsingular term, $\vec{n}_j(\vec{R}-\vec{\rho})/|\vec{R}-\vec{\rho}|$, whose magnitude is bounded by unity. This integral is accurately evaluated by Gauss quadrature. The remaining coefficients in (13) are treated in a similar fashion.

C. Fast Multipole Method. Evaluation of the $4N^2$ influence coefficients, $A_{k,ij}$, is time-consuming. Moreover, during the solution process these coefficients must be accessed several

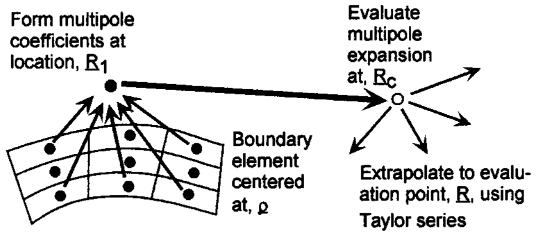
Form multipole coefficients at location, R1
Evaluate multipole expansion at, Rc
Boundary element centered at, ρ
Extrapolate to evaluation point, R, using Taylor series

Figure 1. Schematic showing the location of source and field points relative to the origin of the coordinate system.

times, so in the interest of reduced computation time, it is advantageous to store them after the first evaluation. The storage then scales as $4N^2$ so that even at the modest resolution of $N =$ 10 000 the total storage is 1.6 GB (for single precision), which exceeds the in-core memory capacity of most present-day workstations. To reduce these computational bottlenecks and address large systems, recourse is made to fast multipole methods. These methods combine two basic components. The first is a hierarchical grouping procedure that assembles the BEs into nested groups of specified size. Because of its ability to readily accommodate complex BE assemblies, the adaptive data structure known as an octree[42] is used for this purpose. The octree also facilitates efficient search operations and tests to distinguish between near- and far-field interactions. The second component is a multipole approximation to the electrostatic potential and forces induced by a collection of BEs at sufficiently well-separated, far-field points. The remaining near-field interactions are computed in the conventional way using direct evaluation.

The successful incorporation of fast multipole methods into BEM-based electrostatic models has been demonstrated by several researchers.[33,34,43] However, all past such implementations have been restricted to zero salt concentration ($\kappa = 0$). The reason for this is that whereas multipole expansions have been thoroughly developed for Coulombic potentials, $G_0 = 1/(4\pi r)$, the generalization to screened electrostatic interactions of the form $G_\kappa = e^{-\kappa r}/(4\pi r)$ has only recently received attention. In ref 35, a fast multipole algorithm that formally accommodates solutions to the linear Poisson−Boltzmann equation was described. This algorithm is applicable to both zero ($\kappa = 0$) and finite salt ($\kappa > 0$) conditions and forms the basis of the present fast BEM Poisson−Boltzmann analysis. The following discussion briefly reviews the multipole expansion for the Green's function, $e^{-\kappa r}/r$, and then describes how it is used in the context of the BEM solution of the linear PBE.

Consider (see Figure 1) one of a collection of source points located at position, ρ, and an observation point, R. The relative position vector between the source and observation point is $\vec{r} = \vec{R} - \vec{\rho}$ and the distance is $r = |\vec{r}|$. Upon expressing the position vectors in spherical coordinates, i.e., $\vec{\rho}(\rho,\theta,\phi)$ and $\vec{R}(R,\alpha,\beta)$, a well-known multipole expansion for the Debye−Hückel (screened Coulombic) potential[44] valid for $R > \rho$ may be written as follows:

$$4\pi G_\kappa(\vec{R},\vec{\rho}) = \frac{e^{-\kappa r}}{r} = 8\kappa \sum_{m=0}^{\infty} \hat{I}_{m+1/2}(\kappa\rho)\, \hat{K}_{m+1/2}(\kappa R) \sum_{k=-m}^{m} Y_{mk}(\theta,\phi)\, Y_{mk}^{*}(\alpha,\beta), R > \rho \quad (18)$$

where Y_{mk} is the spherical harmonic function of order m and degree k. The asterisk denotes the complex conjugate, and the spherical modified Bessel functions (SMBFs) of the first and third kind are defined as

$$\hat{I}_{m+1/2}(z) \equiv \sqrt{\frac{\pi}{2z}} I_{m+1/2}(z) \quad (19a)$$

$$\hat{K}_{m+1/2}(z) \equiv \sqrt{\frac{\pi}{2z}} K_{m+1/2}(z) \quad (19b)$$

where $I_{m+1/2}(z)$ and $K_{m+1/2}(z)$ are the corresponding conventional (i.e., not spherical) modified Bessel functions of the first and third kinds, respectively.

It is convenient to define

$$g_m(z) \equiv \frac{1\cdot 3\cdot 5\cdot \ldots \cdot(2m+1)}{z^m} \hat{I}_{m+1/2}(z) \quad (20a)$$

$$G_m(z) \equiv \frac{2}{\pi} \frac{z^{m+1}}{1\cdot 3\cdot 5\cdot \ldots \cdot(2m-1)} \hat{K}_{m+1/2}(z) \quad (20b)$$

$$\Psi_{mk}(\vec{\rho}) = \frac{\rho^m Y_{mk}(\theta,\phi)}{C_{mk}} \quad (21a)$$

$$\psi_{mk}(\vec{R}) = \frac{4\pi}{2m+1} \frac{C_{mk} Y_{mk}(\alpha,\beta)}{R^{m+1}} \quad (21b)$$

$$C_{mk} = \sqrt{\frac{(2m+1)(m-k)!(m+k)!}{4\pi}} \quad (21c)$$

which when substituted into (18) produce

$$4\pi G_\kappa(\vec{R},\vec{\rho}) = \sum_{m=0}^{\infty} g_m(\kappa\rho)\, G_m(\kappa R) \sum_{k=-m}^{m} \Psi_{mk}(\vec{\rho})\, \psi_{mk}^{*}(\vec{R}) \quad (22)$$

This rearrangement is useful in several respects. First, when $\kappa \rightarrow 0$, the small argument behavior of the SMBFs reveals that $g_m(\kappa r) \rightarrow 1$ and $G_m(\kappa R) \rightarrow 1$. Thus

$$\lim_{\kappa\rightarrow 0} 4\pi G_\kappa(\vec{R},\vec{\rho}) = \frac{1}{r} = \sum_{m=0}^{\infty} \sum_{k=-m}^{m} \Psi_{mk}(\vec{\rho})\, \psi_{mk}^{*}(\vec{R}) \quad (23)$$

so that the multipole expansion for the Coulombic potential is recovered. This analytical result is also easily reproduced numerically using the rapidly convergent and stable (with respect to round-off) small argument representations of g_m and G_m. The rescalings for the spherical harmonic functions, (21), are motivated by the simplified integral expressions for Ψ_{mk} and ψ_{mk}^* which facilitate development of recursion relationships and evaluation of these functions in Cartesian coordinates. These functions are also directly related to the so-called "inner" and "outer" functions, $I_m^k(\vec{\rho})$ and $O_m^k(\vec{R})$, employed by White and Gordon[45] and are useful in that there is no need to transform to spherical coordinates to evaluate them. Complete details regarding these functions are given in Appendix A of ref 35.

Another very useful property of the products

$$Q_{mk}(\kappa,\vec{R}) = G_m(\kappa R)\, \psi_{mk}(\vec{R}) \quad (24)$$

is that their spatial gradients can be expressed as linear combinations of Q_{mk} of different order and degree:

$$\nabla Q_{mk} = -\frac{1}{2}\begin{bmatrix} 1 & 0 & -1 \\ i & 0 & i \\ 0 & 2 & 0 \end{bmatrix}\left(\begin{Bmatrix} Q_{m+1,k-1} \\ Q_{m+1,k} \\ Q_{m+1,k+1} \end{Bmatrix} - \frac{\kappa^2}{(2m-1)(2m+1)}[N_{mk}]\begin{Bmatrix} Q_{m-1,k-1} \\ -Q_{m-1,k} \\ Q_{m-1,k+1} \end{Bmatrix}\right) \quad (25a)$$

$$[N_{mk}] = \mathrm{diag}\{(m+k)(m+k-1),\ (m+k)(m-k),\ (m-k)(m-k-1)\} \quad (25b)$$

Recursive application of this relation allows derivatives of arbitrary order, α, to be developed (simply insert appropriate α − 1 order derivatives on the right-hand side of (25a)). Note that if $\kappa \to 0$ then the second term on the right-hand side of (25a) disappears and $Q_{mk} \to \psi_{mk}$ so that the identity for $\nabla\psi_{mk}$ is obtained. The derivations of these relations and additional properties of the SMBFs appear elsewhere (see Appendix B in ref 35).

D. Fast Computation of the Electrostatic Potential. To see how the multipole approximation is used in the context of a fast BEM linear Poisson−Boltzmann equation, consider the electrostatic potential induced by a collection of BEs, $j \in \{G\}$, at a well-separated point, $\vec{R}$. From (5) and (6)

$$\phi_{\{G\}}(\vec{R}) = \sum_{j\in\{G\}} \int_{A_j} G_0(\vec{R},\vec{\rho})h_j + \nabla_R G_0(\vec{R},\vec{\rho})\cdot\vec{n}_j\phi_j \, \mathrm{dA} \quad (26a)$$

$$\mu_{\{G\}}(\vec{R}) = -\sum_{j\in\{G\}} \int_{A_j} G_\kappa(\vec{R},\vec{\rho})\frac{1}{\epsilon}h_j + \nabla_R G_\kappa(\vec{R},\vec{\rho})\cdot\vec{n}_j\phi_j \, \mathrm{d}A \quad (26b)$$

where $h_j = (\partial\phi/\partial n)_j$ and the surface boundary conditions, (4), have been imposed. Substituting for G_κ in (26b) using the multipole expansion, (22), and reordering:

$$4\pi \sum_{j\in\{G\}} \int_{A_j} G_\kappa(\vec{R},\vec{\rho})\frac{h_j}{\epsilon}\,\mathrm{d}A = \sum_{m,k}\left\{\sum_{j\in\{G\}}\int_{A_j} \frac{h_j}{\epsilon}\mathrm{g}_m(\kappa\rho)\Psi_{mk}(\vec{\rho})\,\mathrm{d}A\right\}G_m(\kappa R)\psi_{mk}{}^*(\vec{R}) \quad (27a)$$

$$\nabla_R\cdot\sum_{j\in\{G\}}\int_{A_j} 4\pi G_\kappa(\vec{R},\vec{\rho})\vec{n}_j\phi_j\,\mathrm{d}A = \nabla_R\cdot\sum_{m,k}\{\sum_{j\in\{G\}}\int_{A_j} g_m(\kappa\rho)\Psi_{mk}(\vec{\rho})\phi_j\vec{n}_j\,\mathrm{d}A\}G_m(\kappa R)\psi_{mk}{}^*(\vec{R}) = \sum_{m,k}\{\sum_{j\in\{G\}}\int_{A_j} g_m(\kappa\rho)\Psi_{mk}(\vec{\rho})\phi_j\vec{n}_j\,\mathrm{d}A\}\cdot\nabla_R Q_{mk}{}^*(\kappa,\vec{R}) \quad (27b)$$

In the last identity, (27b), the definition (24) has been used and gradient operator transferred as indicated since the terms inside the brackets {·} are independent of $\vec{R}$. The far-field approximation to (26a) is similarly derived using the limiting form of the expansion, (23), as $\kappa \to 0$. Thus, one obtains

$$\phi_{\{G\}}(\vec{R}) \cong +\sum_{m=0}^{M}\sum_{k=-m}^{m}\{\nabla_R\psi_{mk}{}^*(\vec{R}) \vdots \psi_{mk}{}^*(\vec{R})\}\{B^0_{mk}\} \quad (28a)$$

$$\mu_{\{G\}}(\vec{R}) \cong -\sum_{m=0}^{M}\sum_{k=-m}^{m}\{\nabla_R Q_{mk}{}^*(\vec{R}) \vdots Q_{mk}{}^*(\vec{R})\}\{B^\kappa_{mk}\} \quad (28b)$$

where the expansion has been truncated to M terms to reflect practical implementation. The multipole coefficients for the group of BEs, $j \in \{G\}$, are

$$\{B^0_{mk}\} = \sum_{j\in\{G\}}\int_{A_j}\frac{1}{4\pi}\begin{Bmatrix} n_x\phi \\ n_y\phi \\ n_z\phi \\ h \end{Bmatrix}_j \Psi_{mk}(\vec{\rho})\,\mathrm{d}A \quad (29a)$$

$$\{B^\kappa{}_{mk}\} = \sum_{j\in\{G\}}\int_{A_j}\frac{1}{4\pi}\begin{Bmatrix} n_x\phi \\ n_y\phi \\ n_z\phi \\ h/\epsilon \end{Bmatrix}_j g_m(\kappa\rho)\Psi_{mk}(\vec{\rho})\,\mathrm{d}A \quad (29b)$$

Note that these coefficients are independent of the evaluation point, $\vec{R}$, and can be calculated using only the properties (ϕ_j, h_j, and the element geometry) of the BEs in the group, $\{G\}$. The integrals in (29) involve well-behaved nonsingular functions and are evaluated using Gauss quadrature.

Having established the multipole expansions for a group of elements, the fast evaluation of the electrostatic potential at any point in space becomes a matter of organizing the calculation to appropriately sum contributions from far- and near-field groups. The far-field contribution from a well-separated group of BEs is given by (28). In general, there will be several such groups, $\{G_\alpha\}$, for which a multipole expansion may be invoked and these groups are identified during a pruned downward search through the octree. Contributions from all remaining BEs not contained in one of these groups must be evaluated directly using the conventional singular integrals. In this way all BE contributions are accounted for, so that from (5) and (6):

$$\phi(\vec{R}) = \sum_{j\notin\{G_\alpha\}}\int_{A_j}\left[h_j G_0(\vec{R},\vec{\rho}) - \phi_j\frac{\partial}{\partial n_j}G_0(\vec{R},\vec{\rho})\right]\mathrm{d}A + \sum_\alpha \phi_{\{G_\alpha\}}(\vec{R}) + \sum_{k=1}^{N_q} q_k G_0(\vec{R},\vec{\rho}_k) \quad (30a)$$

$$\mu(\vec{R}) = -\sum_{j\notin\{G_\alpha\}}\int_{A_j}\left[\frac{1}{\epsilon}h_j G_\kappa(\vec{R},\vec{\rho}) - \phi_j\frac{\partial}{\partial n_j}G_\kappa(\vec{R},\vec{\rho})\right]\mathrm{d}A + \sum_\alpha \mu_{\{G_\alpha\}}(\vec{R}) \quad (30b)$$

The first sum contains contributions from BEs considered near-field to $\vec{R}$—those that are not contained in one of the well-separated groups $\{G_\alpha\}$. The second sum contains the contributions from the far-field groups and are evaluated using the multipole approximation, (28).

The simplest implementation that does not use any Taylor series extrapolations proceeds as follows:

1. Develop an octree about the configuration of BEs and define the hierarchical groups—each group, $\{G_\alpha\}$, is comprised of the BEs whose centroids lie within a particular box or cell, α, of the octree.

2. Compute the multipole coefficients, $B^0{}_{mk}$ and $B^\kappa{}_{mk}$, for each group according to (29).

3. Loop over the evaluation points (usually an element centroid or a charge site), i. For each evaluation point conduct a top-to-bottom search through the octree. Starting with the root cell encompassing the entire domain, descendant cells are recursively searched and processed according to the following conditions:

(a) The cell is far-field—evaluate the multipole evaluation and prune the downward search at this cell (i.e., do not search through any descendant cells).

(b) The cell is near-field and has no descendant cells—loop over the BEs, j, in this cell and evaluate contributions directly.

(c) The cell is near-field but contains descendant cells—continue search through the descendants.

A cell, α, is classified as far-field with respect to evaluation point, $\vec{R}$, if $|\vec{R} - \vec{R}_\alpha| > (1 + e_{sc})\Omega_\alpha$, where $\vec{R}_\alpha$ is the cell center and Ω_α is the group size.

The preceding algorithm formally exhibits $O((M + N) \log N)$ computational cost where M is the number of evaluation points and N is the number of BEs. The computational cost can be reduced further to $O(M + N)$ (not including the cost of constructing the octree) through the use of Taylor extrapolation (also known as inner-to-inner translations). Here the loop over individual evaluation points in step 3 above is replaced by another top-to-bottom octree search. Finally, the charge-induced Coulombic potential appearing in (5), (10), and (11) can also be evaluated using the fast algorithm. In the interest of brevity, the reader is referred to ref 35 for more detailed descriptions of these operations.

E. Inversion for Surface Distributions. The preceding section developed fast evaluation procedures to calculate the electrostatic properties at a point *given* the surface solutions ϕ_i and h_i. In the BEM, however, one must first solve an inverse problem and deduce the surface singularities satisfying (12). To this end, an iterative procedure must be used since in the fast BEM the full influence coefficient matrix is no longer available (far-field contributions are represented by multipole approximations). The simplest method is Gauss−Seidel iteration where the boundary elements, i, are visited in succession and the solutions ϕ_i and h_i are updated before proceeding to the next boundary element. For each element, i, the right-hand side of (12), r_{1i} and r_{2i}, are first computed using the current values of ϕ_j and h_j. The local solution is then updated according to

$$\begin{bmatrix} \frac{1}{2}(1+\epsilon) & -A_{2,ii} \\ -A_{3,ii} & \frac{1}{2}\left(1+\frac{1}{\epsilon}\right) \end{bmatrix} \begin{Bmatrix} \Delta\phi_i \\ \Delta h_i \end{Bmatrix} = \begin{Bmatrix} r_{1i} - \frac{1}{2}(1+\epsilon)\phi_i \\ r_{2i} - \frac{1}{2}\left(1+\frac{1}{\epsilon}\right)h_i \end{Bmatrix} \quad (31a)$$

$$\phi_i \leftarrow \phi_i + \Delta\phi_i, h_i \leftarrow h_i + \Delta h_i \quad (31b)$$

before proceeding to the next boundary element.

To improve convergence further, a generalized minimal residual (GMRES)-based outer iteration strategy is also invoked.[46] One Gauss−Seidel sweep essentially modifies the solution vector $\{\gamma\} \equiv \{\phi,h\}$, from $\{\gamma\}^n$ to $\{\gamma\}^{n+1}$. An error vector, $\{e(\gamma^n)\} = \{\gamma\}^{n+1} - \{\gamma\}^n$ can therefore be defined and the GMRES scheme invoked to solve $\{e(\gamma)\} = \{0\}$ by adjusting $\{\gamma\}$. A description of how GMRES is applied to this equation is given elsewhere[46,47] and will be not be repeated here. The combination of Gauss−Seidel and GMRES iteration produces very efficient convergence rates—typically no more than 10−20 evaluations of the right-hand side are required. More importantly, convergence is virtually unaffected by the number of BEs—a result that is attributed to adopting the numerically well-behaved derivative BEM formulation.[24]

F. Mesh Adaptation. A primary advantage to using a low-order boundary element method is the flexibility it offers for adaptive mesh refinement. This advantage arises because in the piecewise constant representation the surface solution is stored at the element centers rather than at the element nodes (as is typical of higher order treatments). This allows surface elements to be subdivided independently from their neighbors without having to maintain a logically contiguous element-to-element connectivity structure. Mesh adaptation is especially beneficial when some of the interior charges are in close proximity to the surface. One can show[23] that for a single charge located at $\vec{\rho}_k$ near the surface the local surface solution behaves as $\phi(\vec{\rho}) \sim A/|\vec{\rho} - \vec{\rho}_k|$ and $(\partial\phi/\partial n)(\vec{\rho}) \sim B\vec{n}\cdot(\vec{\rho} - \vec{\rho}_k)/|\vec{\rho} - \vec{\rho}_k|^3$. Hence as $\vec{\rho} \rightarrow \vec{\rho}_k$ the surface solution becomes singular. The use of higher order interpolation methods is unlikely to offer computational benefits in this case since they only provide good approximations to solutions that are very smooth. Juffer and co-workers[23] addressed the problem of modeling near surface charges by introducing the technique of charge peak separation. Here the surface solution is separated into a rapidly varying part that is approximated analytically and a smoothly varying part that is obtained by the boundary element method. Though improving accuracy, this approach appears to substantially increase the computational costs and coding complexity (e.g., obtaining accurate surface quadratures of the rapidly varying terms is nontrivial). Spatial adaptation, on the other hand, is very easy to implement in a low-order fast boundary element method, requires no alteration to the basic code except during the initial grid setup, and offers good a priori error control.

In the present implementation, an element, i, is uniformly subdivided if

$$|\vec{R}_i - \vec{\rho}_k| < f_r \Delta s_i \quad (32)$$

where $\vec{\rho}_k$ is a fixed charge position, $\vec{R}_i$ is the element control point, Δs_i is the characteristic element size, and f_r is a user-specified parameter controlling the degree of mesh adaptation. If $f_r = 0$, no mesh refinement takes place; as f_r is increased, mesh refinement is carried out to smaller scales and takes place at more distant locations from the charges. Generally, at least $f_r = 1$ is recommended to ensure that the piecewise constant solution representation does not invite gross error; $f_r > 3$ appears to deliver consistently good accuracy. The criterion (32) is applied recursively to the initial triangulation by sweeping through the elements and subdividing those whose size exceeds the permissible level. The subdivided elements are revisited to see whether they need to be further subdivided. This process continues until no further subdivision takes place.

G. Pre- and Postprocessing and Overall Code Operation. The fast BEM-based treatment of the linear Poisson−Boltzmann equation has been embodied in Continuum Dynamics, Inc.'s FPB (Fast Poisson−Boltzmann analysis) code. To successfully conduct an electrostatic calculation, FPB must be accompanied by (i) a surface mesh triangulation routine and (ii) a file containing the atomic coordinates, charges, and radii for the biomolecule being studied. A typical electrostatics computation exercising these components proceeds as follows. A formatted file containing atomic coordinates, charges, and radii is read in by the FPB code. The atomic positions and radii are then passed to the surface triangulation routine. In our work, the efficient and robust MSMS triangulation routine developed by Sanner and Olson[48] (MSMS is available from http://www.scripps.edu/pub/olson-web/people/sanner/html/msms_home.html) has been adopted for this task. This software generates the molecular surface and creates a pair of files defining the associated triangulation. The FPB code also has the capability to read the appropriate files from other surface triangulation programs such as SMART.[49] The MSMS or SMART files, together with the molecule input file and a run file specifying desired input parameters and outputs, are read in by the FPB software. Upon completion of the calculations, postprocessing operations are carried out. Currently, various energies (Coulombic, reaction field, total electrostatic, etc.) are evaluated together with potentials and fields at the charge sites. Optionally, electrostatic properties can be evaluated at user-specified points and surface electrostatic potential maps can be created in the VRML format.

III. Results

Numerous computational experiments have been carried out to validate the FPB code, assess its computational performance and accuracy, and demonstrate its ability to treat large molecules of biological interest. In the following section, results are first presented for a sphere containing interior charges. Analytical results are available for this case, thus allowing rigorous assessment of accuracy. Calculations for biomolecules—alanine dipeptide, barnase, barstar, survivin, FinO, and a coiled coil tetramer—are then performed and compared against predictions using the finite difference Poisson−Boltzmann codes, UHBD, DelPhi, and GRASP. Of particular interest in these studies is the salt-dependent behavior of the energetics of biomolecules. Quantitative measures of this behavior are provided by the following energies:

$$W_{\mathrm{rf}}(\kappa,\epsilon_2) = \frac{1}{2}\sum_q \phi_{\mathrm{rf}}(\vec{\rho}_k)\mathrm{Q}_k \tag{33a}$$

$$\Delta W_{\mathrm{elec}}^{\mathrm{solv}} = W_{\mathrm{rf}}(\kappa,\epsilon_2) - W_{\mathrm{rf}}(0,1) \tag{33b}$$

$$\Delta\Delta W_{\mathrm{elec}}^{\mathrm{solv}} = \Delta W_{\mathrm{elec}}^{\mathrm{solv}}(\kappa,\epsilon_2) - \Delta W_{\mathrm{elec}}^{\mathrm{solv}}(0,\epsilon_2) \tag{33c}$$

$$\Delta W_{\mathrm{rf}} = W_{\mathrm{rf}}(\kappa,\epsilon_2) - W_{\mathrm{rf}}(0,\epsilon_2) \equiv \Delta\Delta W_{\mathrm{elec}}^{\mathrm{solv}} \tag{33d}$$

which are, respectively, the reaction field energy, electrostatic solvation free energy, and ion contribution to the electrostatic solvation free energy. The last expression shows that $\Delta\Delta W_{\mathrm{elec}}^{\mathrm{solv}}$ is equivalently the ion contribution to the reaction field energy, ΔW_{rf}. The reaction field potential, ϕ_{rf}, is evaluated at the charge sites, $\vec{\rho}_k$, and is defined as the surface integral appearing in (5) (i.e., omitting the last sum in (5) which represents the Coulombic potential). One can show that in the limit of zero salt concentration:

$$\left[\frac{\partial W_{\mathrm{rf}}}{\partial \kappa}\right]_{\kappa=0} = \left[\frac{\partial}{\partial \kappa}(\Delta W_{\mathrm{rf}})\right]_{\kappa=0} = \left[\frac{\partial}{\partial \kappa}(\Delta\Delta W_{\mathrm{elec}}^{\mathrm{solv}})\right]_{\kappa=0} = -332\frac{Q^2}{2\epsilon_2} \tag{34}$$

where Q is the net charge. The conversion factor of 332 assumes that Q has units of e (i.e., protonic charge) κ is expressed in 1/Å and energy is given in units of kcal/mol. The result (34) does not depend on the detailed molecular shape, charge distribution, or solute dielectric constant, ϵ_1. It therefore is useful in providing an independent check upon the accuracy of any linear Poisson−Boltzmann equation solver for general systems.

Unless stated otherwise, in all fast FPB calculations presented here second-order ($M = 2$) multipole expansions and Taylor series approximations were employed, the maximum number of BEs per terminal octree box was set to $m_{\mathrm{b}} = 12$ (i.e., any box containing more than 12 BEs is subdivided into 8 child boxes), and the parameter controlling the extent of the near-field region was set to $e_{\mathrm{sc}} = 2.0$. Calculations performed without Taylor series approximations produced virtually no changes in computed results, but increased CPU time by a factor of about 2.

A. Sphere with Interior Charge. The first configuration considered is that of a unit radius spherical cavity, containing an interior unit positive charge, surrounded by aqueous solution. Since analytical expressions for the solution of the linear Poisson−Boltzmann equation are available for all $\kappa > 0$,[50,51] this case constitutes a useful benchmark for establishing the overall accuracy and performance of the FPB code. Two sets of results, one examining the CPU and storage trends and the other assessing errors, are presented.

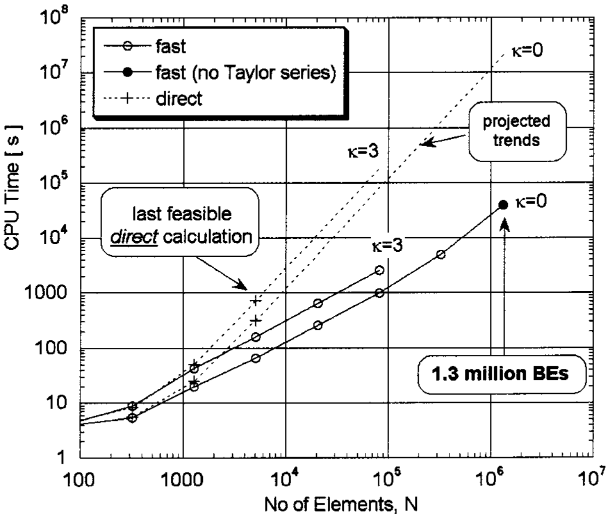

Figure 2. Computation times required to compute the electrostatic potential of the unit spherical cavity containing a centrally located unit charge, immersed in an aqueous solution. The dielectric constants $\epsilon_1 = 2$ and $\epsilon_2 = 80$. Timings were obtained upon a Silicon Graphics single R10000 processor operating at 180 MHz.

For the first set of calculations, a unit charge is located at the center of the sphere and the dielectric constants are set to $\epsilon_1 = 2$ and $\epsilon_2 = 80$. Results for $\kappa = 0$ (Poisson equation) and $\kappa = 3$ (linear PBE) were obtained using different resolution surface grids obtained by recursively subdividing an icosahedron. For comparison, direct computations using low BE counts were also carried out. Note that Gauss−Seidel + GMRES iteration rather than full inversion was used in the direct computation since it is much faster and gives practically identical results. The CPU times (corresponding to a Silicon Graphics single processor R10000 operating at 180 MHz) required to complete the calculation are recorded in Figure 2. Due to memory constraints, it was not possible to perform direct computations using more than 5120 BEs. Nevertheless, the CPU time closely follows a quadratic scaling with N so that reliable projections of CPU time can be made. These projections reveal that for $N = 10^5$ an approximately 2 orders of magnitude speed up is achieved using the fast multipole technology. Also, the CPU time for the fast BE-based PBE solver closely adheres to the theoretically anticipated $O(N \log N)$ behavior. The number of GMRES iterations required to converge the solution is virtually independent of problem size. At all resolutions, convergence to machine round-off was accomplished in three GMRES iterations each involving four fast potential evaluations. This substantiates the earlier comments regarding the excellent numerical conditioning afforded by the derivative BEM formulation.[24]

The greatest performance gains achieved by adopting a fast BEM Poisson−Boltzmann algorithm are in the storage requirements. In conventional BEM PBE analyses, memory costs normally hinder large-scale calculations well before CPU times become prohibitive. Figure 3 depicts the number of stored near-field influence coefficients required for an electrostatic calculation involving N boundary elements. Since this number dominates the total storage, it constitutes a good measure of overall memory requirements. For direct summation, the number of influence coefficients is given exactly by $(pN)^2$, where $p = 1$ if $\kappa = 0$ and $p = 2$ when $\kappa > 0$. The plot confirms that multiple order of magnitude reduction in memory requirements are made

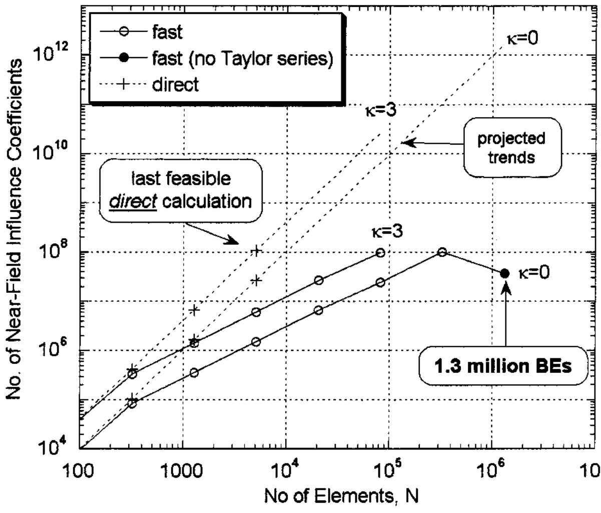

Figure 3. Number of near-field influence coefficients required in the calculation of the electrostatic potential of the unit sphere containing a centrally located unit charge. The dielectric constants are given in the legend of Figure 1.

possible by the FPB analysis. If the maximum number of near-field influence coefficients allowed is 10^8, then FPB extends the maximum number of elements from 4000 to 90 000. It then becomes possible to address correspondingly larger molecules and their complexes or to conduct higher surface resolution calculations than previously possible with the conventional BEM.

To illustrate the ability to perform truly large-scale calculations using the fast BEM PBE solver, the results for a 1.3 million BE calculation are also shown. Unlike the previous results, this calculation did not employ the Taylor series approximation; also the octree subdivision criterion for this case $m_b = 7$. As is evident from Figure 3, these alterations reduced the total number of near-field influence coefficients, which rendered the computation feasible on our machine. These modifications also increased CPU time slightly as seen in Figure 2. The ability to accommodate $O(10^6)$ elements in a Poisson−Boltzmann calculation is of great value when contemplating the study of very large biologically important molecules.

The idealized spherical geometry was used next to assess accuracy. We varied the distance ρ_{charge} of the charge from the center of the sphere and monitored the relative error, $E_{\text{rf}} \equiv (W_{\text{rf}})_{\text{comp}}/(W_{\text{rf}})_{\text{exact}} - 1$, where W_{rf} is the reaction field energy. In general, there are several potential sources of error such as surface quadrature, idealization of the surface geometry, truncation of the multipole series, and approximation of the surface solution. Space limitations do not allow an in-depth examination of all these errors. Instead, we seek to (i) show that the fast multipole approximation does not adversely affect overall accuracy, (ii) demonstrate how mesh adaptation provides a very cost-effective approach for accurately modeling molecular electrostatics, and (iii) identify the dominant error contributors. Figure 4 shows how E_{rf} varies with charge location. The relative error is quite low and behaves in a predictable manner. The relative errors incurred when using fast and conventional summation of the element interactions are similar, which implies that the errors due to truncation of the multipole expansions are small compared to other error sources. The discretization error associated with finite size elements controls overall accuracy, especially for the coarser surface meshes. This can be seen upon inspecting how the errors vary with surface resolution when the charge is placed at the center. From our experience in calculating the electrostatic properties of actual biomolecules, it seems that discretization errors dominate in general. Only when the resolution becomes very fine do multipole series truncation errors become significant (relative to truncation error). Thus, initially, accuracy is best improved by using increasingly finer meshes to reduce discretization error. A point is eventually reached, however, where the truncation errors in the multipole approximation become important. Further reductions in mesh spacing must then be accompanied by an increase in the number of terms retained in the multipole series and/or an increase in the extent of the near-field region.

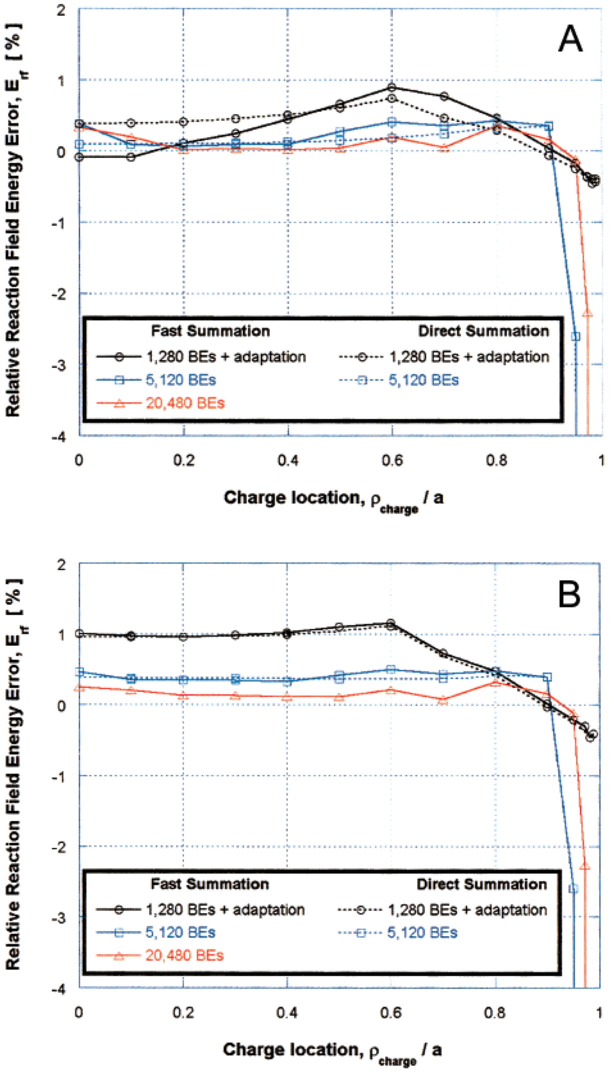

Figure 4. Variation of reaction field energy error, E_{rf}, with the position of a single charge placed at ρ_{charge}/a (where a is the radius of the sphere) in the interior of a unit radius sphere embedded in aqueous solution. The dielectric constants $\epsilon_1 = 1$ and $\epsilon_2 = 80$. The Debye−Hückel parameter (a) $\kappa = 0$ and (b) $\kappa = 3$.

In nonadaptive mesh calculations, accuracy deteriorates as the charge approaches the surface since the mesh cannot adequately resolve the increasingly rapid variations in surface potential induced by the charge. For the $N = 5120$ boundary element discretization, E_{rf} remains below 1% until $\rho_{\text{charge}}/a = 0.9$ (a = sphere radius); for the $N = 20\,480$ element model the same accuracy was sustained up to $\rho_{\text{charge}}/a = 0.95$. In both cases the height above the surface where the error began to grow rapidly is about $h = 2\Delta s$, where the characteristic panel size $\Delta s = (4\pi/N)^{1/2}$. To demonstrate the effectiveness of adaptive meshing, a calculation was performed using an initial discretization of 1280 BEs which were then adaptively refined according to (32) with $f_r = 4$. The error is now seen to remain small even when the charge is very close to the surface. For $\rho_{charge}/a = 0.985$ (the closest charge−surface interaction considered) the number of elements in the final adapted mesh was only $N = 1892$ and the relative error $E_{rf} < 0.5\%$. By comparison, a nonadaptive mesh calculation of comparable accuracy would require approximately 900 000 elements. Clearly, mesh adaptation offers a powerful approach for achieving high accuracy at low computational cost.

We make a final observation on a charge that is placed well away from the surface. The error on the calculated electrostatic potential is closely related to how well the numerical method represents the distance between the charge and the surface. If one triangulates the surface of a unit sphere using N flat elements whose vertices lie on the sphere, then one can show that the centroid of each element lies at a radius of approximately $1 - (4.8/N)$. Thus, for a central charge, the potential induced by the charge at the element control points differs from the exact value by about $4.8/N$. For the 1280 BE discretization, this difference is 0.4%, so it is not surprising that the relative error, E_{rf}, is on this order. Moreover, in this case the error reduces only inversely with N so that a 10-fold error reduction warrants a 10-fold increase in the number of elements. This problem could be simply rectified by using flat elements whose vertices lie on a slightly larger sphere (or, alternatively, by the use of curvilinear elements). As the charge approaches the surface, the energy sum is dominated by contributions from the nearest surface elements and curvature effects are relatively unimportant. The surface potential now varies rapidly, and mesh adaptation is the most effective means of improving overall accuracy.

B. Electrostatic Solvation Free Energies of Alanine Dipeptide. As a validation exercise for the FPB code, the classical alanine dipeptide molecule was considered. Many studies have examined the conformational dependence of the solvation free energy of alanine dipeptide, which is a neutral molecule with 22 atoms (e.g.,ref 52). It has been shown that, due to their larger net dipole moments, the helical-like conformations (αR and αL) of alanine dipeptide have more favorable solvation free energies compared to the C7ax, C7eq, and C5 conformers.[52] Also, a theoretical study by Tazaki and Doi[53] shows that the conformational stability of different conformations of alanine dipeptide is affected by the addition of salt. Here, the electrostatic component of the solvation free energy is computed for different conformers of alanine dipeptide at zero added salt and 0.1 M NaCl, using the FPB, DelPhi, and UHBD programs.

The geometries of the different alanine dipeptide conformers (C7eq, C5, C7ax, αR) were taken from the study of Scarsdale et al.[54] The CHARMM22 molecular mechanical force field[55] was used to assign atomic charges and radii. The solvent-excluded surface (SES) (using a 1.4 Å probe radius) was employed to define the dielectric interface between the solute and solvent regions. The dielectric constants were taken as $\epsilon_1 = 1$ (solute) and $\epsilon_2 = 80$. The solution temperature was 298 K. For the DelPhi calculations a 201^3 grid with 12.5 grid points/Å was employed and no focusing was invoked. For the UHBD calculations a two-step focusing protocol was employed with grid sizes and spacings of 140^3 and 0.5 Å (first step) and 140^3 and 0.1 Å (second step). In both DelPhi and UHBD codes, the potential at each boundary point was set using the Debye−Hückel approximation and no dielectric boundary smoothing was employed.

Table 1 summarizes the computation times and electrostatic solvation free energy predictions obtained with the FPB program using different BE densities (with $f_r = 0$). These results indicate that the energies are essentially converged using $N = 4328$ elements (i.e., higher element counts produce less than 1% variations in computed energy) or about 200 BEs/atom. The results reported in Table 2 are obtained with this resolution. The number of GMRES iterations required to obtain a converged solution ranged between 4 and 5 for all surface resolutions considered in Table 1. The computation time essentially varies linearly with the number of BEs, thus confirming that the theoretical CPU trends expected for the fast BEM are indeed obtained in practice.

TABLE 1: Dependence of the Electrostatic Solvation Free Energy of the C5 Alanine Dipeptide Conformer (at 0.1 M NaCl) on the Surface Resolution[a]

no. of boundary elements	$\Delta W^{solv}{}_{elec}$ (kcal/mol)	CPU time (s)
394	−16.90	5.9
638	−13.20	10.2
1210	−12.54	29.8
2336	−12.22	52.2
4328	−12.14	129.9
7954	−12.09	239.4
1338	−12.06	480.2
32856	−12.04	1007.5
67212	−12.04	4590.0

[a] The timings were obtained on a Silicon Graphics single R10000 processor operating at 180 MHz.

TABLE 2: Electrostatic Solvation Free Energies (in kcal/mol) of Four Conformers of Alanine Dipeptide (C7eq, C7ax, C5, and αR), Obtained Using Different Poisson−Boltzmann (PB) Algorithms[a]

PB algorithm	C7eq	C7ax	C5	αR
FPB ($N = 4328$ BEs)	−10.63	−10.60	−12.14	−13.85
	(−10.62)	(−10.60)	(−12.13)	(−13.84)
UHBD	−10.31	−10.42	−11.95	−13.66
	(−10.29)	(−10.41)	(−11.93)	(−13.63)
UHBD[b] (no added salt)	−9.85	−9.75	−11.00	
DelPhi (no added salt)	−10.59	−10.56	−12.09	−14.12

[a] The solute and solvent dielectric constants were 1 and 80, respectively. The ionic strength was either 0 or 0.1 M. The number in parentheses corresponds to no added salt. [b] These results were taken from Schaefer et al.[63]

Table 2 compares the electrostatic solvation free energy of all four conformers of alanine dipeptide obtained using the FPB program and finite difference based PBE algorithms. All PBE codes predict very similar relative ΔW^{solv}_{elec} for the conformers of alanine dipeptide, based on the CHARMM22 force field parameters and exactly the same alanine dipeptide geometries. The prediction of a more favorable electrostatic solvation free energy for the αR conformer relative to the other conformations is in accord with published results.[52,56] Good agreement of FPB results with the UHBD and DelPhi results is also observed. However, there is a pronounced difference between our results at zero added salt and those found in an earlier work obtained by UHBD.[63] This is probably due to differences in the geometries of the alanine dipeptide conformers, the use of dielectric boundary smoothing, and different grid spacings employed in these UHBD calculations. As shown in Table 2 for this neutral molecule, addition of salt practically does not affect the electrostatic solvation free energy of the alanine dipeptide conformers. In summary, all PBE codes here considered predict the correct relative electrostatic solvation free energies of the alanine dipeptide conformers. However, contrary to the Tzaki and Doi study, which considered the AMBER molecular mechanical force field, FPB and UHBD results

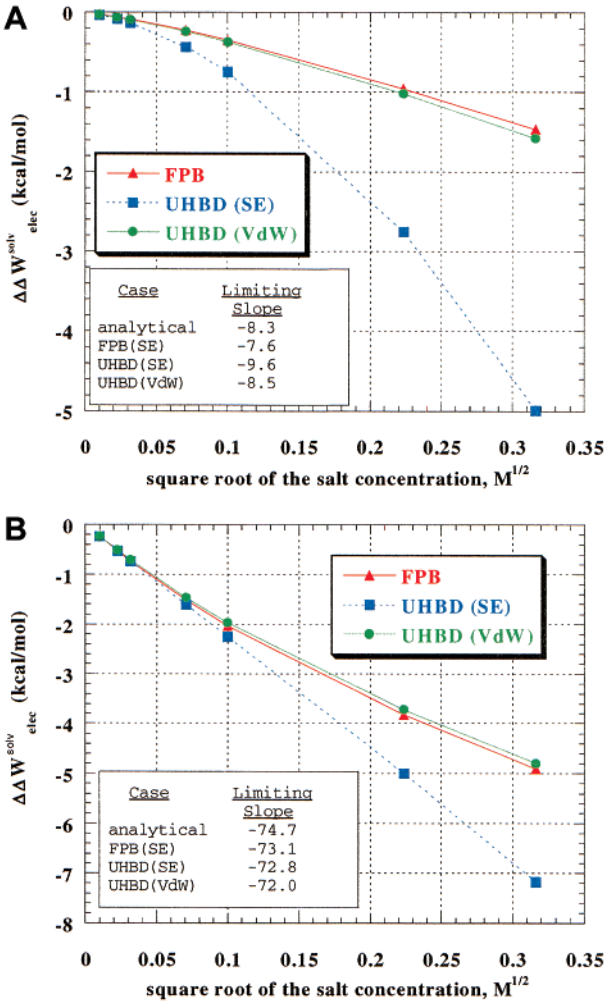

Figure 5. Ionic strength dependence of $\Delta\Delta W^{\mathrm{solv}}_{\mathrm{elec}}$ for (a) barstar and (b) barnase obtained using both FPB and UHBD programs. Both the van der Waals (vdW) and solvent-excluded (SE) molecular surfaces are employed to define the dielectric boundary. The limiting low-salt slopes are in units of kcal Å/mol.

suggest that the conformational preferences of the alanine dipeptide are not affected by the addition of salt.

For these small molecules, accurate results are obtained using relatively low BE counts. These calculations could, in fact, easily be performed without the fast multipole algorithm. In the remaining sections, larger numbers of boundary elements are necessary to properly model the molecular geometry and produce accurate results. Fast multipole technology is essential in these calculations since without it the associated CPU and storage penalties would be prohibitive.

C. Salt Dependence of the Electrostatic Solvation Free Energy of Barnase and Barstar. Electrostatic interactions are believed to play an important role in the biological function of charged proteins. Some studies have shown that salt effects are significant in the barnase−barstar association process.[57−59] In this section, the salt dependence of the proteins—wild-type barnase (net charge of $+2e$) and its tight inhibitor barstar (net charge of $-6e$)—are considered. The protocol for preparing the systems and the assignment of parameters closely follow those in earlier studies. The atomic coordinates of barnase and barstar were taken from the X-ray structure of the barnase−barstar complex[60] and the missing residues and hydrogen atoms added with the InsightII program (Accelrys, Inc). The structures were then subjected to constrained energy minimization with the heavy atom positions held fixed. The charges and radii of the atoms were assigned from the AMBER molecular mechanical force field.[61] The proteins and solvent were assigned dielectric constants of 4.0 and 80.0, respectively, and the temperature of the aqueous NaCl solution was set to 298 K. No Stern layer was considered.

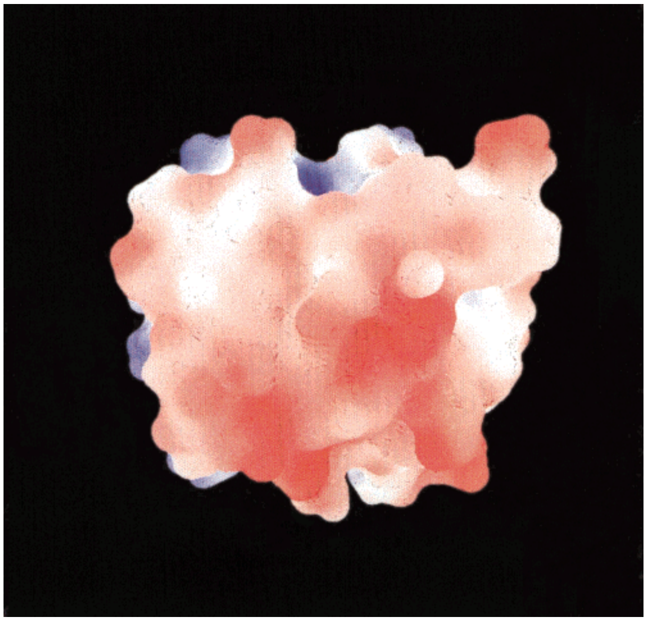

Figure 6. Surface electrostatic potential of barstar. The calculation was done assuming a uniform dielectric constant of 80 for the solvent and 2 for the protein interior and physiological ionic strength (i.e., 100 mM NaCl). The color of the surface represents the electrostatic potential at the protein surface, going from red (potential of $-6.8\ k_{\mathrm{B}}T/e$) to blue (potential of $+6.8\ k_{\mathrm{B}}T/e$); white is neutral potential.

The solvent-excluded (SE) molecular surface (1.4 Å probe radius) was adopted in the FPB calculations. The initial triangulation of these surfaces was developed with the MSMS program followed by mesh adaptation with $f_{\mathrm{r}} = 2$. The final number of BEs is $N = 81\,597$ for barnase and $N = 65\,656$ for barstar. Such element counts are well beyond the means of conventional BEM-based PBE algorithms. For the UHBD calculations a two-step focusing protocol was employed. The grid size and grid spacing were 140^3 and 2.0 Å in the first step and 140^3 and 0.35 Å in the second run. The boundary potential was set using the Debye−Hückel approximation. UHBD results were obtained using both the van der Waals (vdW) and solvent-excluded surfaces, the latter being obtained by setting the UHBD code parameters nmap = 1.4 Å (probe radius) and nsph = 500 (number of surface points per atom).

Figure 5 depicts the variation of $\Delta\Delta W^{\mathrm{solv}}_{\mathrm{elec}}$ with salt concentration for barnase and barstar. The plots clearly show a more pronounced salt dependence of $\Delta W^{\mathrm{solv}}_{\mathrm{elec}}$ for the molecule with higher net charge as is expected from (34). Physically, this means that the transfer of a charged molecule from the gas phase to the aqueous solution phase is more favorable when the net charge of the molecule is higher and when salt is added to the aqueous solution phase. According to (34), at sufficiently low NaCl concentrations $\Delta\Delta W_{\mathrm{elec}}$ varies linearly with κ and hence with the square root of the salt concentration. Both UHBD and FPB programs predict similar slopes for the charged proteins at low salt concentrations. At 0.0001 M NaCl, the respective slopes obtained by finite differencing with the $\kappa = 0$ results are -7.6 (FPB), -8.5 (UHBD + vdW surface), and -9.6 (UHBD + SE surface) kcal Å/mol. These compare well with the analytical slope of -8.3 kcal Å/mol. For the more highly charged protein, the limiting low-salt slopes are -73.1 (FPB),

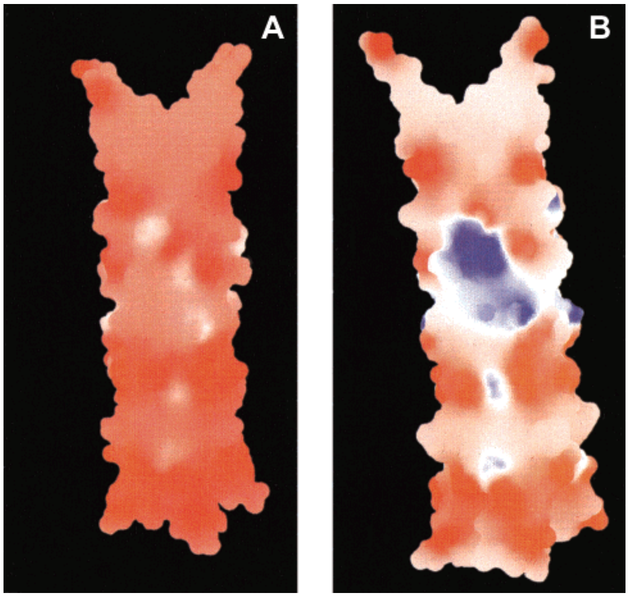

Figure 7. Surface electrostatic potential of the parallel right-handed coiled coil tetramer at zero added salt (a) and 100 mM NaCl (b), respectively. The scale is (−10.2,0,+0.5) for (a) and (−5.1,0,+3.4) for (b), respectively, in GRASP notation and units of k_BT/e, where negative values are red and positive values are blue.

−72.0 (UHBD + VdW), −72.8 (UHBD + SE), and −74.7 (analytical) kcal Å/mol. Both FPB and UHBD results show that the low-salt slope agrees with the analytical result (34) independent of the molecular surface and interior dielectric constant (results not shown). At higher salt concentrations the curves obtained with FPB and UHBD + vdW diverge from the UHBD + SES results. The salt dependence of the ΔW_{elec}^{solv} of barnase and barstar obtained using the FPB code was reproduced with an alternate finite difference PBE solver (results not shown). Also, similar behavior was found using either the vdW or SE surfaces. This is in contrast to the UHBD results, which show that the salt dependence of these two molecules is significantly affected by the choice of molecular surface.

D. Surface Electrostatic Potential of Protein Molecules. Most of the biological activity of a biomolecule lies on its surface. The mapping of the electrostatic potential on the molecular surfaces of biomolecules is now widely used in the identification of potential ligand binding and functional sites and in determining electrostatic complementarity (e.g., in protein−protein complexes). In the new era of computational proteomics, with the increasing number and size of available high-resolution X-ray crystallography and NMR biomolecular structures, fast and accurate Poisson−Boltzmann solvers that generate color-coded surface electrostatic potential maps and other surface-based electrostatic properties (e.g., induced surface charge density) are essential. The boundary element approach used in the FPB code provides high accuracy for such surface electrostatic potential and induced surface charge density maps since these electrostatic properties at each surface element is provided directly, without the need for any interpolation scheme, such as required in the finite difference PBE approaches. The FPB software provides VRML files containing the electrostatic potential and induced surface charge density patterns on the molecular surface.

Here, as a validation exercise we examine the surface electrostatic potential of barstar and three elongated protein molecules: parallel right-handed coiled coil tetramer (PDB entry: 1FE6; net charge = −20*e*, 3279 atoms), FinO (PDB entry: 1DVO, net charge = +6*e*, 2481 atoms), survivin (PDB entry: 1F3H, net charge = −6*e*, 4450 atoms). The input parameters are the same as used for barstar in section III.C above. The solvent molecules, ions, and other ligand molecules in the crystal structures were removed. The hydrogen atoms were added (using a pH value of 7) with the InsightII program (Accelrys, Inc.), and the final structures were subjected to a 1000 cycle conjugate gradient energy minimization (based on the AMBER force field) using the Discover module implemented in InsightII (Accelrys, Inc.). A formal charge set parameter was used in the Poisson−Boltzmann calculations. The van der Waals radii were taken from the GRASP[62] default radii parameter set: H = 1 Å; C = 1.7 Å, O = 1.6 Å; N = 1.65 Å. The dielectric boundary between the proteins and solvent was defined by the solvent-excluded surface using a 1.4 Å solvent probe. The MSMS program was used to generate the triangular elements. The number of elements after mesh adaptation (f_r = 2) was 117 541(parallel right-handed coiled coil tetramer), 101 794 (FinO), and 145 690 (survivin). The ionic strength of the solution was either zero or 100 mM NaCl, and the temperature of the solution was 298 K. The interior dielectric constant was 2, and an exterior value of 80 was employed.

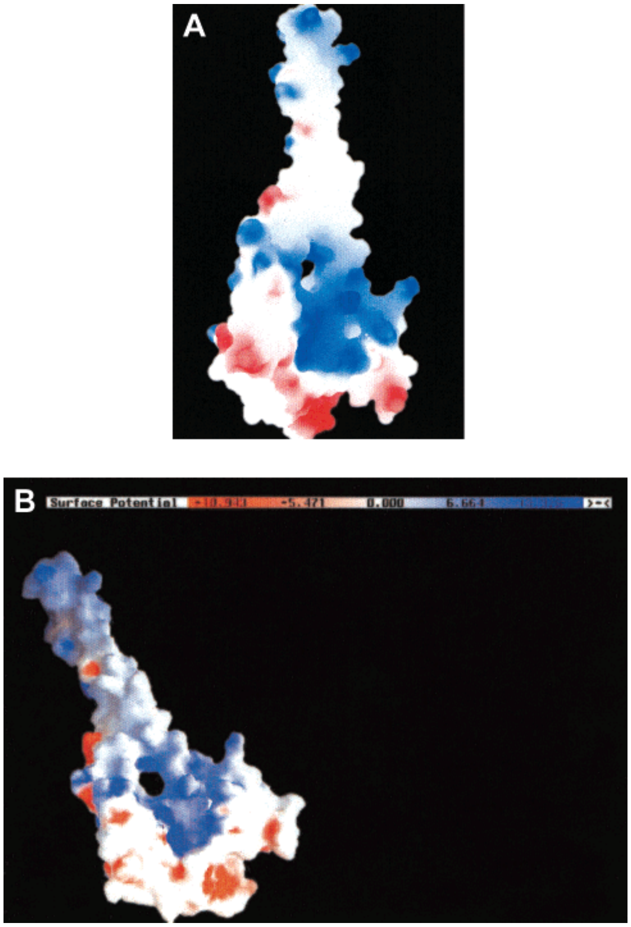

Figure 8. Mapping of the electrostatic potential (in k_BT/e = 0.59 kcal/mol/*e*) on the surface of the FinO basic protein at 100 mM NaCl, computed with the FPB (a) and GRASP (b) programs, respectively. The scale is (−6.8,0,+6.8) for (a) and (−10.9,0,+13.3) for (b), respectively, in GRASP notation.

The globular and negatively charged barstar molecule has a large region of negative electrostatic potential due to the large number of negatively charged residues (see Figure 6). Figure 7 compares the electrostatic potential on the surface of the highly negatively charged parallel right-handed coiled coil tetramer calculated under the conditions of no added salt and 100 mM

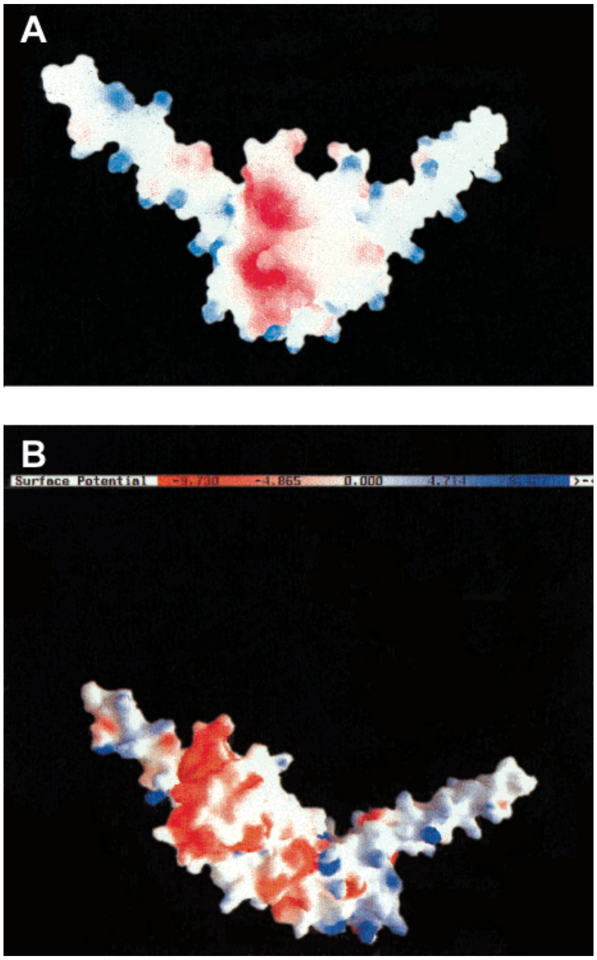

Figure 9. Electrostatic potential surface (in k_BT/e) rendering of survivin at 100 mM NaCl (a). The GRASP surface electrostatic potential is shown in (b). Regions colored blue and red indicate negative and positive electrostatic potentials, respectively. The scale is (−13.6,0,+10.2) for (a) and (−9.7,0,+9.4) for (b), respectively, in GRASP notation.

NaCl. At zero salt concentration the surface electrostatic potential is very negative over most of the molecular surface due to the large number of acidic residues. The screening effect of the salt produces smaller absolute electrostatic potentials and regions of positive electrostatic potential appear. Next, we compare the surface electrostatic potential of the basic FinO RNA binding protein computed with the FPB and GRASP linear PBE solvers at 100 mM NaCl. As shown in Figure 8 both Poisson−Boltzmann codes predict similar regions of both positive and negative electrostatic potential. Both PBE programs generate two distinct regions of positive electrostatic potential (one in the central domain and one in the end of the index finger of the right-hand fist motif of this protein) that could be regions where the negatively charged RNA molecule could bind to the protein. Last, the surface electrostatic potential of a larger elongated acidic survivin protein obtained with the FPB program is compared with that computed with the GRASP program. In Figure 9 we observe that due to the asymmetric distribution of acidic and basic residues there are many distinct regions of both positive and negative electrostatic potential. Both FPB and GRASP electrostatic potential maps are in good qualitative agreement, with both programs predicting a large region of negative electrostatic potential in the BIR domains (shown protruding from the center of the molecule; the BIR domains are located in the region where the two α-helices come together), where zinc ions bind.

IV. Conclusions

A fast multipole algorithm has been successfully incorporated into a boundary element method to solve the Poisson−Boltzmann equation. By selecting the appropriate multipole expansion, problems involving finite salt concentrations can be accurately addressed with the method, thus generalizing previous fast boundary element based treatments that were limited to zero salt concentration. A derivative formulation[23] is adopted to ensure good numerical conditioning at large element counts and is shown to sustain fast convergence rates irrespective of problem size. A simple, yet effective adaptive capability has been incorporated to ensure adequate resolution and allow efficient modeling of problems involving near-surface charges. Application of the approach to the idealized configuration of a sphere containing interior charges has demonstrated good accuracy and confirmed the anticipated $O(N \log N)$ CPU and storage trends. For typical meshes, the errors introduced by the multipole approximation are found to be negligible compared to those incurred by the use of finite size elements and approximate surface distributions. Demonstration calculations involving more than 1 million elements have also been performed to illustrate the ability to address truly large systems with a fast boundary element method. Finally, the fast multipole-accelerated Poisson−Boltzmann solver has been used to study the salt dependence of electrostatic properties of several biomolecules. This study has shown that the fast BEM-based Poisson−Boltzmann solver accurately predicts energies, surface potentials, and the variations of these properties with salt concentration for realistic biomolecular configurations.

Acknowledgment. This work has been supported by the NIH under Grant 5 R44 GM57764-03 and GM58187 (to H.-X.Z.). We are grateful to Dr. Michael Sanner and Prof. Randy Zauhar for providing their molecular triangulation programs (MSMS and SMART).

References and Notes

(1) Darden, T. A.; York, D. M.; Pedersen, L. G. *J. Chem. Phys.* **1993**, *98*, 10089−10092.

(2) Sagui, C.; Darden, T. A. *Annu. Rev. Biophys. Biomol. Struct.* **1999**, *28*, 155−179.

(3) Cheatham, T. E. I.; Kollman, P. A. *Annu. Rev. Phys. Chem.* **2000**, *51*, 435−471.

(4) Penfold, R.; Warwicker, J.; Jonsson, B. *J. Phys. Chem. B* **1998**, *102*, 8599−8610.

(5) Gilson, M. K.; Honig, B. H. *Nature* **1987**, *330*, 84−86.

(6) Demchuk, E.; Wade, R. C. *J. Phys. Chem.* **1996**, *100*, 17373−17387.

(7) Juffer, A. H.; Vogel, H. J. *Proteins: Struct. Funct. Genet.* **2000**, *41*, 554−567.

(8) Northrup, S. H.; Boles, J. O.; Reynolds, J. C. L. *J. Phys. Chem.* **1987**, *91*, 5991−5998.

(9) Radic, Z.; Kirchhoff, P. D.; Quinn, D. M.; McCammon, J. A.; Taylor, P. *J. Biol. Chem.* **1997**, *272*, 23265−23277.

(10) Gogonea, V.; Merz, K. M. J. *J. Chem. Phys.* **2000**, *112*, 3227−3235.

(11) Kollman, P. A.; Massova, I.; Reyes, C.; Kuhn, B.; Huo, S.; Chong, L.; Lee, M.; Lee, T.; Duan, Y.; Wang, W.; Donini, O.; Cieplak, P.; Srinivasan, J.; Case, D. A.; Cheatham, T. E. I. *Acc. Chem. Res.* **2000**, *33*, 889−897.

(12) Warwicker, J. *J. Theor. Biol.* **1986**, *121*, 199−210.

(13) Klapper, I.; Hagstrom, R.; Fine, R.; Sharp, K.; Honig, B. *Proteins: Struct. Funct. Genet.* **1986**, *1*, 47−59.

(14) Bashford, D.; Karplus, M. *Biochemistry* **1990**, *29*, 10219−10225.

(15) Cortis, C. M.; Friesner, R. A. *J. Comput. Chem.* **1997**, *18*, 1591−1608.

(16) Davis, M. E.; Madura, J. D.; Luty, B. A.; McCammon, J. A. *Comput. Phys. Commun.* **1991**, *62*, 187−197.

(17) Madura, J. D.; Briggs, J. M.; Wade, R. C.; Davis, M. E.; Luty, B. A.; Ilin, A.; Antosiewicz, J.; Gilson, M. K.; Bagheri, B.; Scott, L. R.; McCammon, J. A. *Comput. Phys. Commun.* **1995**, *91*, 57−95.

(18) Nicholls, A.; Honig, B. *J. Comput. Chem.* **1991**, *12*, 435−445.

(19) Holst, M.; Saied, F. *J. Comput. Chem.* **1993**, *14*, 105−113.

(20) You, T. J.; Harvey, S. C. *J. Comput. Chem.* **1993**, *14*, 484−501.

(21) Holst, M.; Baker, N.; Wang, F. *J. Comput. Chem.* **2000**, *21*, 1319−1342.

(22) Baker, N.; Holst, M.; Wang, F. *J. Comput. Chem.* **2000**, *21*, 1343−1352.

(23) Juffer, A. H.; Botta, E. F. F.; van Keulen, B. A. M.; van der Ploeg, A.; Berendsen, H. J. C. *J. Comput. Phys.* **1991**, *97*, 144−171.

(24) Liang, J.; Subramaniam, S. *Biophys. J.* **1997**, *73*, 1830−1841.

(25) Pratt, L. R.; Tawa, G. J.; Hummer, G.; Garcia, A. E.; Corcelli, S. A. *Int. J. Quantum Chem.* **1997**, *64*, 121−141.

(26) Purisima, E. O.; Nilar, S. H. *J. Comput. Chem.* **1995**, *16*, 681−689.

(27) Rashin, A. A.; Namboodiri, K. *J. Phys. Chem.* **1987**, *91*, 6003−6012.

(28) Vorobjev, Y. N.; Scheraga, H. A. *J. Comput. Chem.* **1997**, *18*, 569−583.

(29) Yoon, B. J.; Lenhoff, A. M. *J. Comput. Chem.* **1990**, *11*, 1080−1086.

(30) Zauhar, R. J.; Morgan, R. S. *J. Comput. Chem.* **1988**, *9*, 171−187.

(31) Zhou, H.-X. *Biophys. J.* **1993**, *65*, 955−963.

(32) Greengard, L.; Rokhlin, V. *Chem. Scr.* **1989**, *29A*, 139−144.

(33) Zauhar, R. J.; Varnek, A. *J. Comput. Chem.* **1996**, *17*, 864−877.

(34) Purisima, E. O. *J. Comput. Chem.* **1998**, *19*, 1494−1504.

(35) Boschitsch, A. H.; Fenley, M. O.; Olson, W. K. *J. Comput. Phys.* **1999**, *151*, 212−241.

(36) Richards, F. M. *Annu. Rev. Biophys. Bioeng.* **1977**, *6*, 151−176.

(37) Vijayakumar, M.; Zhou, H.-X. *J. Phys. Chem. B* **2001**, *105*, 7334−7340.

(38) Zauhar, R. J.; Morgan, R. S. *J. Comput. Chem.* **1990**, *11*, 603−622.

(39) Oosterom, A. v.; Strackee, J. *IEEE Trans. Biomed. Eng.* **1983**, *30*, 125−126.

(40) Munck, J. C. d. *IEEE Trans. Biomed. Eng.* **1992**, *39*, 986−990.

(41) Ferguson, A. S.; Zhang, X.; Stroink, G. *IEEE Trans. Biomed. Eng.* **1994**, *41*, 455−460.

(42) Samet, H. *The Design and Analysis of Spatial Structures*; Addison-Wesley Publishing Company, Inc.: Reading, MA, 1990.

(43) Bharadwaj, R.; Windemth, A.; Sridharan, S.; Honig, B.; Nicholls, A. *J. Comput. Chem.* **1995**, *16*, 898−913.

(44) Abramowitz, M.; Stegun, I. A. *Handbook of Mathematical Functions*; U.S. Department of Commerce: Washington, D.C., 1968; Vol. 55.

(45) White, C. A.; Head-Gordon, M. *J. Chem. Phys.* **1994**, *101*, 6593−6605.

(46) Wigton, L. B.; Yu, N. J.; Young, D. P. GMRES Acceleration of Computational Fluid Dynamics Codes; AIAA Paper 85-1494. *Proceedings of the AIAA Conference*, Denver, CO, 1985.

(47) Boschitsch, A.; Curbishley, T. B.; Quackenbush, T.; Teske, M. E. A Fast Panel Method for Potential Flows About Complex Geometries. Presented at the 34th AIAA Aerospace Sciences Meeting & Exhibit, Reno, NV, 1996; Paper A96-0024.

(48) Sanner, M. F.; Olson, A. J. *Biopolymers* **1996**, *38*, 305−320.

(49) Zauhar, R. J. *J. Computer-Aided Mol. Des.* **1995**, *9*, 149−159.

(50) Sader, J. E.; Lenhoff, A. M. *J. Colloid Interface Sci.* **1998**, *201*, 233−243.

(51) Kirkwood, J. G. *J. Chem. Phys.* **1934**, *2*, 351−361.

(52) Brooks, C. L. I.; Case, D. A. *Chem. Rev.* **1993**, *93*, 2487−2502.

(53) Tazaki, K.; Doi, J. *J. Phys. Chem.* **1996**, *100*, 14520−14525.

(54) Scarsdale, J. N.; Van Alsenoy, C.; Klimkowski, V. J.; Schafer, L.; Momany, F. A. *J. Am. Chem. Soc.* **1983**, *105*, 3438−3445.

(55) MacKerell, A.; Bashford, D.; Bellott, M.; Dunbarack, R.; Evanseck, J.; Kuchnir, L.; Kuczera, K.; Lau, F.; Mattos, C.; Michnik, S.; Ngo, T.; Nguyen, D.; Prodhom, B.; Reiher, W.; Roux, B.; Smith, J.; Stote, R.; Straub, J.; Watanabe, M.; Wiorkiewicz-Kuczera, J.; Yin, D.; Karplus, M. *J. Phys. Chem. B* **1998**, *102*, 3586.

(56) Yamane, T.; Inoue, Y.; Sakurai, M. *Chem. Phys. Lett.* **1998**, *291*, 137−142.

(57) Vijayakumar, M.; Wong, K.-Y.; Schreiber, G.; Fersht, A. R.; Szabo, A.; Zhou, H.-X. *J. Mol. Biol.* **1998**, *278*, 1015−1024.

(58) Selzer, T.; Schreiber, G. *J. Mol. Biol.* **1999**, *287*, 409−419.

(59) Gabdouline, R. R.; Wade, R. C. *Biophys. J.* **1997**, *72*, 1917−1929.

(60) Buckle, M.; Schreiber, G.; Fersht, A. R. *Biochemistry* **1994**, *33*, 8878−8889.

(61) Weiner, S. J.; Kollman, P. A.; Case, D. A.; Singh, U. C.; Ghio, C.; Alagona, G.; Profeta, S.; Weiner, P. *J. Am. Chem. Soc.* **1984**, *106*, 765−784.

(62) Nicholls, A.; Sharp, K.; Honig, B. *Proteins: Struct. Funct. Genet.* **1991**, *11*, 281−296.

(63) Schaefer, M.; Karplus, M. A Comprehensive Analytical Treatment of Continuum Electrostatics. *J. Phys. Chem.* **1996**, *100*, 1578.