Performance Comparison of Generalized Born and Poisson Methods in the Calculation of Electrostatic Solvation Energies for Protein Structures

MICHAEL FEIG,[1] ALEXEY ONUFRIEV,[1] MICHAEL S. LEE,[2] WONPIL IM,[1] DAVID A. CASE,[1] CHARLES L. BROOKS III[1]

[1]*Department of Molecular Biology, The Scripps Research Institute, 10550 North Torrey Pines Road, La Jolla, California 92037*

[2]*Department of Cell Biology and Biochemistry, USAMRIID, 1425 Porter St., Frederick, Maryland 21702*

Received 23 June 2003; Accepted 15 July 2003

Abstract: This study compares generalized Born (GB) and Poisson (PB) methods for calculating electrostatic solvation energies of proteins. A large set of GB and PB implementations from our own laboratories as well as others is applied to a series of protein structure test sets for evaluating the performance of these methods. The test sets cover a significant range of native protein structures of varying size, fold topology, and amino acid composition as well as nonnative extended and misfolded structures that may be found during structure prediction and folding/unfolding studies. We find that the methods tested here span a wide range from highly accurate and computationally demanding PB-based methods to somewhat less accurate but more affordable GB-based approaches and a few fast, approximate PB solvers. Compared with PB solvation energies, the latest, most accurate GB implementations were found to achieve errors of 1% for relative solvation energies between different proteins and 0.4% between different conformations of the same protein. This compares to accurate PB solvers that produce results with deviations of less than 0.25% between each other for both native and nonnative structures. The performance of the best GB methods is discussed in more detail for the application for force field-based minimizations or molecular dynamics simulations.

 J Comput Chem 25: 265–284, 2004

Key words: continuum dielectric; implicit solvation

Introduction

A complete energetic description of biomolecular processes has to include the aqueous solvent environment that is present *in vivo* and *in vitro*.[1,2] Theoretical models of such systems may include solvent explicitly or implicitly. Explicit solvent representation, where a biological molecule of interest is embedded in a fairly large number of solvent molecules, is straightforward and can provide a realistic description of energetic and kinetic effects.[3] However, substantial computational resources are already required for single protein chains and many larger systems such as protein oligomers and macromolecular assemblies will remain prohibitively expensive in the foreseeable future. Implicit solvent models, on the other hand, replace explicit solvent interactions with an equivalent energetic term based on its mean field behavior.[4–6] Such an approach can be computationally much more affordable while still providing a realistic description of an aqueous solvent environment.

Continuum Solvent Description Based on Poisson Theory

The most popular models for implicit solvation begin from a continuum electrostatic viewpoint and assume that the macroscopic description of solvent as a continuous dielectric medium can be used as an approximation on the microscopic scale.[7,8] The physical system for calculating the electrostatic contribution of the energetic effects of solvation for a given molecule can then be reduced to a distribution of charges in a solvent-inaccessible low dielectric cavity surrounded by a homogeneous high dielectric media. The electrostatic potential $\phi(r)$ in such a model is described rigorously by the Poisson equation:

$$\nabla[\varepsilon(r)\nabla\phi(r)] = -4\pi\rho(r) \quad (1)$$

where $\rho(r)$ and $\varepsilon(r)$ are the space-dependent charge distribution and dielectric constant, respectively. If this equation is expanded to include ionic salt concentrations it is called the Poisson–Boltzmann (PB) equation (in linear and nonlinear forms). Because we are not considering salt effects in this study we are concerned only with solutions to the simpler Poisson equation, but we will use the common abbreviation PB nevertheless for convenience and tradition.

Correspondence to: C. L. Brooks III; e-mail: brooks@scripps.edu

If a classic force field is used to describe molecular interactions, the charge distribution is given by the partial charges located at the atomic centers and the dielectric boundary most commonly follows the molecular surface, which is defined by the contact points of a sphere of the size of the solvent molecule with the van der Waals surface of the molecule of interest.[9] While the choice of the external dielectric constant depends on the solvent media (e.g., $\varepsilon = 80$ for water), the choice of the internal dielectric constant has been the subject of some debate. While a value of 1 is appropriate if simulations are used to sample conformational fluctuations explicitly, values from 2 or 4 or higher have been suggested for evaluating solvation energies for static structures, when dynamic contributions to the dielectric response are not directly included[7,10–12] and the choice of a single dielectric constant assuming that the protein interior can be treated as a dielectric continuum may be problematic altogether.[12]

The molecular surface definition provides a sharp dielectric boundary between the solute cavity and the external dielectric. A more gradual change of $\varepsilon(r)$ across the boundary may be more realistic, and a smooth boundary also facilitates the calculation of continuous forces for atoms at the dielectric boundary upon conformational changes during simulations.[13–15] Nevertheless, the simple two-dielectric system based on the Lee–Richards molecular surface of the molecule of interest has remained the *de facto* standard for the calculation of solvation energies with a continuum solvent model because it models directly the effective solvent accessibility based on a spherical solvent molecule approximation.

The PB equation can be solved numerically in different ways.[16] Quite popular are finite difference methods, where charges and dielectric constants are discretized using a grid.[17] PB solvers based on finite difference methods such as UHBD[18] or DelPhi[19,20] have been available for some time and enjoy widespread use. High-accuracy results can be achieved with sufficiently fine grids, but the computations can become costly and memory intensive for increasing system sizes. In fact, memory limitations are usually more prohibitive than the CPU time that would be required for solution of the equations. Improvements have focused on multigridding and focusing techniques that offer advantages for larger system sizes.[21,22]

Alternative approaches are given by finite element methods[21–24] and boundary element methods[25–27] where explicit polarization charges are calculated at the solvent boundary. The electrostatic potential is then computed from the interaction of the molecular charges with these polarization charges. Recently the boundary element method has been combined with fast multipole methods, resulting in better scaling than finite difference methods with increasing system size.[28,29]

Once the electrostatic potential is known, the electrostatic energy is given by

$$G_{\text{elec}} = \frac{1}{2}\int_V \rho(r)\phi(r)dV \tag{2}$$

where the volume is integrated over all space. The electrostatic free energy of solvation is then given as the difference between G_{elec} in vacuum (i.e., with an external dielectric of 1) and G_{elec} in solution:

$$\Delta G^{\text{el}}_{\text{solvation}} = G^{\varepsilon=80}_{\text{elec}} - G^{\varepsilon=1}_{\text{elec}} \tag{3}$$

or for a set of discrete charges, $\{q_i\}$

$$\Delta G^{\text{el}}_{\text{solvation}} = \frac{1}{2}\sum_i q_i(\phi^{\varepsilon=80}(r_i) - \phi^{\varepsilon=1}(r_i)) \tag{4}$$

Normally, application of eq. (4) requires the PB equation to be solved twice, once in vacuum ($\varepsilon = 1$) and a second time for the desired solution environment (e.g., $\varepsilon = 80$). It is possible, however, to calculate the induced polarization surface charges from the electrostatic potential near the dielectric boundary and then calculate the electrostatic solvation energy from a simple Coulomb interaction between the solute charges and the polarization charges, as in the boundary element method, without requiring a second finite-difference run.[20] In addition, this approach also provides more accurate results with coarser grid sizes because the grid is only used for solving the PB equation while the location of the induced surface charges can be interpolated between grid points.

PB calculations are commonly used to evaluate electrostatic potentials for static structures of biomolecules in solution.[2] PB models have also been successful in the determination of pK_a values of protonizable chemical groups to account for pH effects on binding and stability of biomolecules.[30–33] More recently, PB calculations have been used on larger structural ensembles to provide electrostatic free energies of solvation in scoring applications following the MMPB/SA scheme.[34–36] The application of PB solutions for molecular dynamics simulations is more limited due to high computational costs and technical difficulties. However, a number of studies have demonstrated that implicit solvent simulations that use the PB equation directly can be feasible.[37–44]

Approximation of Poisson Theory Through Generalized Born Formalisms

Because of the significant computational expense of numerical solution of the PB equation, many efforts have focused on equivalent energetic descriptions that describe the same or similar continuum solvent at a much reduced cost.[7] The most popular approach in this respect is based on the generalized Born (GB) formalism.[45] The starting point in this model is the Born solvation energy for a single charge q in a sphere of radius R that is embedded in a medium with the dielectric constant ε:[46]

$$\Delta G^{\text{el}}_{\text{solvation}} = -\frac{q^2}{2R}\left(1 - \frac{1}{\varepsilon}\right) \tag{5}$$

For a polyatomic system occupying a more complex shape this has been generalized by an ansatz based on a pairwise sum over interacting charges:[47,48]

$$\Delta G^{\text{el}}_{\text{solvation}} = -\frac{1}{2}\left(1 - \frac{1}{\varepsilon}\right)\sum_{i,j}\frac{q_i q_j}{\sqrt{r_{ij}^2 + \alpha_i\alpha_j\exp(-r_{ij}^2/F\alpha_i\alpha_j)}} \tag{6}$$

where r_{ij} is the distance between atoms i and j, q_i and q_j are the respective (partial) charges, and α_i are the effective or so-called

generalized Born radii, which may be interpreted roughly as the distance from each atom to the dielectric boundary. For the factor F a value of 4 has been proposed originally.[48] Extensions of this model to include salt effects have been proposed.[49]

Key in the successful application of the GB model is the definition and calculation of α_i, the Born radii. The Born radius for a given atom can be calculated exactly by solving the PB equation assuming a unit charge for the atom while the rest of the molecule is uncharged but still present to define the dielectric boundary. It has been demonstrated that if these atomic PB radii are used as the α_i in eq. (6) the resulting solvation energies correspond surprisingly well to the PB solvation energies for the entire molecule.[50] Of course, following this prescription would require much more work to solve the PB equation to obtain values for α for each atom. The real advantage of the GB formalism then comes from a formulation of efficient means to calculate atomic Born radii, α_i, for a given configuration without having to solve the PB equation.

The basic idea for the estimation of Born radii is based on the so-called Coulomb field approximation (CFA), which is exact for a charge in the center of a spherical cavity and assumes that the dielectric displacement follows a Coulombic form, independent of the external dielectric.[45] Inverting eq. (5) and expressing ΔG in terms of the dielectric displacement results in the following expression for α_i (for a complete derivation see Ref. 45):

$$\frac{1}{\alpha_i} = \frac{1}{R_i} - \frac{1}{4\pi} \int_{\text{solute}, r>R_i} \frac{1}{r^4}\, dV \tag{7}$$

where R_i are the atomic radii (e.g., van der Waals radius) used to define the solute cavity filling out the volume V, over which the integral is calculated. This is the basis for most GB implementations, although the somewhat costly evaluation of the integral is often avoided by using a discrete sum over atomic volumes with an appropriate overlap function for computational convenience.[51–56] Such approaches can be implemented efficiently and are suitable for simulations because of their analytic nature. A related variation of this scheme is based on an approximation of the solute volume by overlapping Gaussian functions as in the ACE model.[57,58] Another variation has been proposed in the form of an alternative surface integral formulation, with the motivation to find improved agreement with PB solvation energies.[59] More recently, higher-order correction terms were added to eq. (7) as heuristic corrections beyond the CFA, resulting in Born radii that were found to be in near-perfect agreement with PB radii and, consequently, more accurate estimates of total solvation energies when the integral form was evaluated numerically.[60,61] (A differentiable form of the numerical volume integration can be maintained for the use in molecular dynamics simulations.[60]) This improved accuracy in reproducing the PB model comes at somewhat higher costs compared to other GB implementations.

GB models have been used successfully for calculating accurate solvation energies of small molecules,[51,59,62] scoring of protein conformations,[63–65] evaluation of protein–ligand binding,[66] pK_a predictions,[54] and implicit solvent molecular dynamics simulations.[55,66–69] However, problems in achieving stable trajectories of native conformations when using GB for simulations have also been pointed out.[70–72]

The number of GB implementations has been growing rapidly, but only limited comparisons of GB methods and between GB and PB methods, in terms of accuracy and speed, have been published to date.[70,71,73] Such a study might be in particular helpful in directing users of GB methods to find the most suitable version for different types of applications. A comparative study will also serve as a guide to future development efforts. It is the goal of this study to provide such information for a number of commonly used GB implementations. Obviously, many aspects are important in evaluating implicit solvent models. However, we intend to focus here only on the ability of different GB methods to reproduce electrostatic free energies of solvation in the context of continuum models as described by the PB equation because both are based on the same underlying physical model. While a comparison of GB models with explicit solvent simulations and experiment is equally important, such comparisons will not be discussed here. To put the performance of GB methods into perspective, we have also included a number of commonly used PB methods as yardsticks in terms of accuracy and speed. While a comparison of various GB methods between themselves as well as an evaluation of their performance relative to PB results is the goal of this study, we are not attempting to compare the performance of various PB methods. Criteria that determine the latter are often nontrivial (e.g., convergence and grid sizes) and the question of reference energies becomes more critical. For the purposes of this study we used solvation energies that were obtained by finite difference solutions to the Poisson equation with a fine grid spacing of 0.25 Å as reference values for both GB and PB methods.

Much recent interest in implicit solvation techniques has revolved around their application in simulations of biologic macromolecules, so that longer timescales or larger system sizes can be modeled compared to approaches that consider solvent explicitly. While many GB methods have been evaluated extensively for small molecules,[51,59,62] we will reflect the interest in modeling macromolecules by using a large number of representative native and nonnative protein structures as test cases.

In the next section we will describe the GB and PB implementations that are being compared in this study. Then we will present the results we obtained in terms of accuracy and speed, discuss our findings with respect to potential applications, and finally summarize our conclusions.

Methods

Test Sets

Five different test sets were used in this study to cover different aspects of protein structure conformations (see Table 1). For all of the structures we generated complete all-atom representations as in the CHARMM22,[74] Amber94,[75] and OPLSAA force fields.[76] The initial structures were used as is, without any minimization or refinement, and with missing hydrogen atoms added if necessary using the HBUILD procedure in CHARMM.[77] Explicit solvent molecules, ions, or any coligands that may have been present in experimental structures were not included in the calculations. All of the test structures are available from the authors upon request.

Table 1. Summary of Test Sets Used in This Study.

Test set	Structures	Residues	RMSD (Å)
1: Training	22	30–98	0
2: PDB	611	30–839	0
3: Villin headpiece	120	36	3.2–15.3
4: Protein L	216	62	2.8–14.1
5: Myoglobin (pH = 2.0)	101	153	3.5–28.4

For each test set the number of structures, range of number of residues, and range of root mean square displacements (RMSDs) from the experimental native structure are given. The PDB codes for test sets 1 and 2 are given in the Appendix.

Set 1 contains 22 native protein structures from the Protein Data Bank (PDB)[78] with less than 100 residues that were used as the initial training set for some of our GB methods. The PDB codes for the structures in this set are given in the Appendix.

Set 2 contains a comprehensive set of 611 nonhomologous single-chain PDB structures ranging from small protein fragments to very large structures with more than 800 residues and covering a wide variety of native folds. The PDB codes are also given in the Appendix. This set presents a more challenging test than set 1 for the calculation of solvation energies in native protein structures. Because of the very large size of some of the structures in this set we could not run all PB methods for all structures given present memory and time limitations. Some GB methods also did not result in valid solvation energies for some structures in this test set, apparently due to the erroneous prediction of negative Born radii by some of the approaches.

Sets 3, 4, and 5 were used to test how well solvation energies for different native and nonnative conformations of the same protein can be reproduced with different GB and PB methods. Set 3 consists of 120 near-native, misfolded, and unfolded structures of chicken villin headpiece (PDB code 1VII). These conformations were generated through a lattice sampling protocol with low-resolution representations[79] followed by a subsequent reconstruction of all-atom models.[80] Set 4 consists of 216 different conformations for protein L (PDB code 2PTL) that were generated by thermal unfolding in explicit solvent, and set 5 contains 101 representative structures of apomyoglobin from a recent study of this molecule's unfolding pathway.[81] In set 5 the structures are also protonated according to an acidic environment with pH = 2 and consequently have a much higher charge than typical proteins at neutral pH values. All of the PB and GB methods were tested with set 3, but only a subset of methods was applied to sets 4 and 5.

Reference Solvation Energies

Reference solvation energies were calculated with the standard finite difference method implemented in the PBEQ module[14,82,83] in CHARMM.[77] A grid spacing of 0.25 Å was deemed sufficient for the purposes of this comparison while approaching the limits of feasibility for the largest molecule in the test sets. The molecular surface was used to define the dielectric boundary based on a spherical water probe radius of 1.4 Å. A margin of at least 4.5 Å from the extent of each structure to the edge of the grid was allowed in all cases to avoid boundary effects. Charges were distributed onto grid points using the trilinear interpolation method and successive overrelaxation[84] was used to speed convergence, which was reached in all calculations. For the boundary potential a simple Coulomb term was calculated on every other grid point and interpolated for grid points in between. As summarized in Table 2, different sets of solvation energies were calculated for a number of combinations of charges and atomic radii to be able to compare PB and GB methods that are based on different force fields and/or sets of radii. While most PB methods are flexible in this respect by allowing atomic charges and radii as external input, many GB implementations are tied more closely to specific force fields and/or radii through parameterization or lack of input options. If arbitrary charges and radii could be used, the charges and van der Waals radii from the CHARMM22 force field were chosen

Table 2. Reference Energy Sets Calculated from Finite Difference Solutions to the PB Equation with 0.25-Å Grid Spacing Using the PBEQ Module in CHARMM.

Reference set	Charges	Radii	ε_{int}	ε_{ext}
CHARMM	CHARMM22[a]	CHARMM22[a]	1	80
Amber94Bondi	Amber94[b]	Bondi[c]	1	80
Amber94mBondi	Amber94[b]	modified Bondi[d]	1	80
Amber94Eps2	Amber94[b]	Amber94[b]	2	80
OPLSImpact	OPLS[e]	OPLS/Impact[f]	1	80
CHARMMSmooth[g]	CHARMM22[a]	CHARMM22[a]	1	80

[a]Standard CHARMM22 charges and van der Waals radii.[74]
[b]Standard Amber 94 charges and van der Waals radii.[75]
[c]van der Waals radii derived by Bondi[90]: C*, 1.7 Å; N*, 1.55 Å; O*, 1.5 Å; H*, 1.2 Å; S*, 1.8 Å.
[d]Modified Bondi radii, where $H^{C/N}$ = 1.3 Å and H^{0} = 0.8 Å.[52,67]
[e]Standard OPLS charges.[76]
[f]Modified OPLS radii, where H = 1.0 Å if zero.
[g]Smooth boundary definition.[14]
For each set the charges, atomic radii, and dielectric constants are given.

by default. We also calculated a set of reference energies with a smooth, van der Waals-based surface as implemented in the PBEQ module.[14] In all cases except one (see Table 2), an internal solute dielectric of 1 and an external solvent dielectric of 80 were used. Salt effects were not considered here because many GB implementations and some PB methods do not offer this option.

GB Methods

We tested various GB implementations in the latest versions of the CHARMM, Amber, Tinker, and Impact programs. These comparisons are described in detail below.

CHARMM

The CHARMM program,[77] version c30a1, implements four different GB and GB-like methods. The first version, called cGB^{STILL}, follows a pairwise approach developed by Still and coworkers for the calculation of Born radii[51] but uses a slightly modified linearized version to facilitate parameterization by fitting to solvation energies from PB calculations.[55] For the tests performed here we used the original published parameters with the λ factor recommended for proteins for the CHARMM22 force field. We also increased atomic radii of polar hydrogens to 0.8 Å (from a value of 0.2 Å given by the force field) as recommended.[55]

The second model, called GBMV, is a newly developed GB implementation that calculates Born radii by analytic volume integration according to eq. (7).[61] It includes a higher-order correction term to the Coulomb field approximation:

$$A_7 = \left(\frac{1}{4R_i^4} - \frac{1}{4\pi} \int_{\text{solute}, r>R_i} \frac{1}{r^7}\, dV \right)^{1/4} \tag{8}$$

which yields an improved fit between GB and PB radii. This version also incorporates an analytic formalism for approximating the molecular surface rather than the van der Waals surface that would result from a simple overlap of atomic spheres.[60] This method is tunable in terms of accuracy and speed by the choice of integration grid points. In this study we used the following grid sizes: 26-point and 38-point Lebedev grids[85,86] as well as regular angular grids with N_ϕ values of 5 and 8 (for a detailed description see Ref. 61). We used the published parameters in the latest article,[60] including a modified value of 8 for parameter F in eq. (6) instead of 4. We also examined a recently improved empirical adjustment of calculated Born radii based on solvation energies for all of the structures in test set 2. In the original parameterization the function

$$\alpha_i = \frac{S}{\left(1 - \frac{1}{\sqrt{2}}\right) A_4 + A_7} + D \tag{9}$$

is used to calculate the final Born radii with A_7 defined as in eq. (8) and A_4, the regular Coulomb field contribution, given in eq. (7). The parameters S and D were optimized for the CHARMM22 force field to $S = 0.9085$ and $D = -0.102$.[60] The new parameterization uses the more general expression

$$\alpha_i = \frac{S}{C_0 A_4 + C_1 A_7} + D \tag{10}$$

which was found to produce a better fit than eq. (9) with $S = 0.9114$, $C_0 = 0.2966$, $C_1 = 1.0369$, and $D = -0.0637$. These parameters appear to be more universally applicable and were used for the calculation of solvation energies based on the Amber force field as well.

A slower, grid-based, nonanalytic version of GBMV is also available in CHARMM as described by Lee et al..[61] Because we found that the results are much the same as with the analytic method when angular grids with $N_\phi = 8$ are used this method is not included in the present study.

The third model, termed GBSW, follows much of the same methodology used in GBMV but without its expensive molecular surface approximation.[87] Instead, a van der Waals-based surface with a smooth dielectric boundary, as introduced earlier for calculation of PB solvation forces,[14] is modeled in GBSW. Simply by setting S to 1 and D to 0, C_0 and C_1 in eq. (10) were optimized for various smoothing lengths against the exact Born radii of a small protein (1AJJ in test set 1) calculated by PB with the smooth boundary. In this study, we used a smoothing length of 0.6 Å (i.e., $w = 0.3$ Å) along with 38 Lebedev angular integration points and 24 radial integration points up to 20 Å for each atom.[87] The integration points and weights for the radial component were generated by Gaussian–Legendre quadrature[88] and those for the angular component by Lebedev quadrature.[86] We also tested an alternative parameterization, $GBSW^{MS}$, which targets electrostatic solvation energies obtained for a regular, sharp molecular surface rather than smooth van der Waals-based surfaces in the context of GBSW. For this purpose, we reoptimized C_0 and C_1 in eq. (10) for a smoothing length of 0.4 Å (i.e., $w = 0.2$ Å) with test set 1 and the best agreement was found with $C_0 = 1.204$ and $C_1 = 0.187$. The solvation energies were calculated with $F = 4$ in eq. (6).

A fourth version is the GB-like model ACE by Schaefer and Karplus,[57] as reparameterized recently for the all-atom version of the CHARMM force field.[58] It uses a precomputed set of atomic volumes that are accumulated using Gaussian functions to represent the molecular volume. We tested the two published parameter sets based on the Voronoi volumes for the CHARMM22 force field with and without atomic hydrogen volumes, called ACE22 and ACE22h, respectively. Both of these parameters require a slight extension to the original CHARMM22 topology and parameter files to allow ACE to distinguish aromatic carbon atoms as well as amino and imino nitrogen atoms in arginine side-chains. As recommended, we used a Gaussian width of $\alpha = 1.3$ for the parameters without hydrogens and $\alpha = 1.5$ for the parameters that include hydrogen volumes. We also tested a set of new parameters with zero hydrogen volumes, called ACE22n, that do not require a modified CHARMM22 force field with $\alpha = 1.3$ (Michael Schaefer, personal communication). All of the tests with ACE were done with the ACE2 code in CHARMM version c30a1.

Amber

The molecular mechanics package Amber 7 also implements a number of GB methods. All of the methods in Amber are essentially based on the pairwise descreening formulation of the calculation of Born radii by Hawkins et al.[56,89] In the first version, aGB^{HCT}, this method was reparameterized for use with the Amber force field and shown to perform quite well on small molecules and DNA.[52,67]

A second model, aGB^{OBC} is based on aGB^{HCT} (to preserve its performance on small compounds) but better approximates the molecular volume of larger compounds by introducing an appropriate rescaling procedure[53] for the effective Born radii calculated from van der Waals spheres in a modified version of eq. (7):

$$\frac{1}{\alpha_i} = \frac{1}{R_i - s} - \frac{1}{R_i}\tanh(\delta\psi - \beta\psi^2 + \gamma\psi^3) \tag{11}$$

where $s = 0.09$ Å and ψ is given by

$$\psi = \frac{R_i - s}{4\pi}\int_{r>R_i}\frac{1}{r^4}\,dV \tag{12}$$

Two parameterizations, aGB^{OBC}(I) and aGB^{OBC}(II), with the parameters $\delta = 0.8$, $\beta = 0$, and $\gamma = 2.91$ (I) and $\delta = 1.0$, $\beta = 0.8$, and $\gamma = 4.85$ (II) have recently been applied in MD simulations [Onufriev, Bashford, and Case: Exploring protein native states and large-scale conformational changes with a modified generalized Born model. (submitted)] and will be tested here as well.

Two more GB versions are available in Amber that follow the same approach for the calculation of Born radii as aGB^{HCT} but apply modified versions of eq. (6) for the calculation of solvation energies from Born radii.[62] In the first of these variants, aGB^{JSB}, only the exponential prefactor F in eq. (6) is changed from 4 to 2. The second variant, aGB^{MGB}, uses a different functional form derived previously from the consideration of simple model systems.[54] Both aGB^{JSB} and aGB^{MGB} assume an internal dielectric of 2, which will be taken into account when comparing to PB solvation energies. All of the GB methods in Amber were tested with the charges from the Amber force field but different sets of radii as recommended for each method. The van der Waals radii from the Amber force field were only used for aGB^{JSB} and aGB^{MGB}. In aGB^{OBC}(I) and aGB^{OBC}(II) the Bondi radii set[90] was used instead. For aGB^{HCT} the radii for carbon, nitrogen, oxygen, and sulfur atoms were taken from Bondi as well but radii for hydrogen atoms were modified slightly.[52]

Tinker

We tested two GB methods from the Tinker molecular dynamics package, version 3.9.[91] The first one, tGB^{STILL}, directly implements the pairwise summation proposed by Qui et al.[51] The second method, tGB^{HCT}, directly follows the pairwise descreening approach by Hawkins et al.[56] For the purposes of this comparison we used Tinker with the CHARMM22 force field charges and van der Waals radii.

Impact

The surface integral method, S-GB, developed by Ghosh et al.,[59] was implemented originally in the Impact program.[92] This program is now part of Schrödinger, Inc.'s FirstDiscovery program suite. In this approach, a surface formulation of eq. (7) is obtained by applying Gauss' law and empirical correction terms are added to improve the calculation of solvation energies in agreement with results from PB calculations, in particular for larger molecules. We used the standard set of parameters for S-GB in Impact. Impact uses the OPLS force field[76] but with slightly modified atomic hydrogen parameters for the use in S-GB (all of the zero van der Waals radii from OPLS were set to 1.0 Å).

Poisson Methods

In addition to the GB methods described above we also looked at a number of popular PB solvers. This was done mostly to provide a perspective for the performance of different GB version. The different PB methods that were considered in this study are described below.

CHARMM PBEQ

As mentioned above, the finite difference method from the PBEQ module[14,82] in CHARMM was used to calculate the reference solvation energies. While the reference energies were calculated with a grid spacing of 0.25 Å, we also calculated energies with grid spacings of 0.4 and 0.5 Å to evaluate how speed can be gained at the expense of accuracy with this implementation.

MEAD

MEAD,[93] version 2.2.0, also implements the finite difference method and was included mainly to validate the CHARMM PBEQ results. Solvation energies from MEAD were calculated only for test sets 1 and 3 because the methodologies in MEAD and CHARMM PBEQ are so similar. As with CHARMM, we also calculated solvation energies with a larger grid spacing of 0.5 Å. When running MEAD we used the default setup with an initial grid spacing of 1 Å before focusing down to the desired grid resolution. The size of the cubic grid was adjusted accordingly to fit the molecule with sufficient margin as for the CHARMM PBEQ calculations described above.

DelPhi

The DelPhi program[2,19,20] has been a popular choice for calculating solution electrostatics. While it also uses the finite difference method for solving the Poisson equation, it has the interesting feature of requiring only one instead of two finite difference solution runs. Such savings are achieved by calculating the induced polarization charges directly and deriving the solvation energy from the interaction between the solute charges and the reaction field due to the polarization charges.[20] This approach also allows the use of coarser charge distribution grids, which speeds the calculation even more. In the present study we calculated reaction field-based solvation energies with the latest version, DelPhi V.4, using grid spacings of 0.25, 0.5, and 1.0 Å and a

convergence criterion of 0.1 kT/e. We also tested the older version, II, which is still widely used by many groups, and found no significant difference in terms of accuracy or speed for the calculations of solvation energies without salt.

APBS

The APBS program,[21,22] version 0.2.4, represents a new class of multigrid finite element solvers, where the errors for initial coarse-grid solutions are estimated and used for adaptive refinement at finer resolutions. Three types of surfaces are available with APBS: the molecular surface, a molecular surface with simple harmonic average smoothing, and a spline-based smooth surface following the idea of Im et al.[14] for obtaining continuous forces at the boundary that corresponds to the smooth surface in CHARMM. In all tests we used four hierarchy levels with final grid spacings of 0.40 and 0.50 Å. Depending on the size of the molecule, grid sizes of 65, 97, 129, 161, 193, and 225 points in each dimension were sufficient to fit all of the molecules at 0.50-Å grid spacing and all but a few very large molecules at 0.40 Å. Note that the method limits the number of grid points to $n \cdot 2^5 + 1$ for four hierarchy levels. Consequently, a larger number of points often has to be used compared to finite difference methods to cover the spatial extent of a given molecule. Grid sizes larger than 225^3 were not practical due to memory constraints because memory requirements approached 2 Gb (for 225 grid points). For the same reason we could not test grid spacings smaller than 0.4 Å with APBS on a complete test set. In running APBS we used trilinear charge interpolation (chgm = 0) and Debye–Hückel boundary conditions with superposition for each ion (bcfl = 2) because we found that they gave the best results in our comparison.

Impact PBF

A finite element Poisson solver based on tetrahedral meshes, called PBF, is available in the Impact package from Schrödinger, Inc..[23,24] As with S-GB, the OPLS force field is used to provide charges and radii, except for hydrogen atom radii that were changed to 1.0 Å from 0.0 Å for PB as well as the GB calculations. Although the article describing the method also compares results for a regular molecular surface definition with DelPhi, the implementation in Impact apparently only uses a smooth boundary definition for the calculation of smooth forces[94] (Schrödinger, Inc., technical support, personal communication). Consequently, a comparison of solvation energies from PBF with other PB and GB methods is less straightforward.

REBEL

We also tested the fast boundary element method, REBEL,[95] which is implemented in the ICM software package.[96] For this study we used version 3.0.017 (available from MolSoft, LLC). Due to technical limitations it was not possible to calculate solvation energies with the very small atomic van der Waals radii for polar hydrogens in the CHARMM22 force field in the fast, default mode of REBEL. Instead, we used charges from the Amber force field and Bondi radii, consistent with some of the GB methods implemented in the Amber program. In addition to the default setup, we also calculated solvation energies with a much higher density of surface boundary elements as given by the "exact" option in REBEL. Note that slight modifications to the original program did allow the use of small hydrogen radii with the "exact" option (Max Totrov, MolSoft, LLC, personal communication), but this was not tested extensively.

ZAP

ZAP, obtained from Openeye Software on November 4, 2002, solves the PB equation using a Gaussian-based molecular volume definition. The resulting boundary with a smooth transition from $\varepsilon = 80$ to $\varepsilon = 1$ allows much faster convergence compared to conventional PB methods, but the gain in speed comes at the expense of a less accurate reproduction of the exact molecular surface.[15] We used ZAP with grid spacings of 0.5 and 0.25 Å.

Performance Analysis

The results of the GB and PB methods tested here were evaluated by comparing the calculated solvation energies with the appropriate set of reference energies (see Table 2). Absolute and relative errors with respect to PB solvation energies as well as standard deviations were calculated for each of the five test sets. While absolute solvation energies may be shifted by a constant value in different methods due to grid artifacts or for other reasons, the correct estimate of relative solvation energies between conformations is most important in applications of GB or PB methods. For this reason, we will focus on using the minimal errors from optimally shifted energies for comparing the accuracy of different methods. The constant energy shifts that minimize the error for each data set and each method were determined by heuristic search and are given below as part of the results.

The timing of all methods was done on a Pentium IV (2 GHz) workstation running Linux. The following versions of the packages described above were used for timing: standard distributions of Tinker and Impact; CHARMM version c30a1, compiled with the Portland Group compiler according to the standard CHARMM installation script; Amber, versions 7 and 8beta, the latter implementing recent speedups in the GB routines, compiled with the Intel IFC compiler; and DelPhi, compiled with the Portland Group compiler and optimized at the O2 level.

Results

GB Methods

The comparison of different GB methods is given in Table 3. We find that the overall level of accuracy varies substantially between different implementations and also between test sets. In general, test set 2, which represents a comprehensive set of small and large protein structures, appears to be the most challenging as indicated by the large average errors and root mean square (RMS) deviations in the solvation energy. On the other hand, the errors in the solvation energies for the villin headpiece and protein L in test sets 3 and 4 are fairly small with almost all of the methods. For the myoglobin structures, the percentage errors with respect to the PB

Table 3. Comparison of Electrostatic Solvation Energies from GB Methods with Appropriate PB Reference Energies.

GB method	PB reference	Average/maximum error (%)				
		Set 1 (training)	Set 2 (PDB)	Set 3 (villin)	Set 4 (protein L)	Set 5 (myoglobin)
cGB^{STILL}	CHARMM	6.37/18.3	13.73[a]/99.6	3.65/10.9	3.16/9.6	1.76/4.1
GBMV[b]	CHARMM	1.21/3.2	1.02/5.5	0.46/1.9	0.55/2.1	0.09/0.3
GBMV[c]	CHARMM	1.14/3.3	0.82/7.2	0.40/1.6	0.50/1.8	0.09/0.3
GBMV[d]	CHARMM	1.24/3.9	0.82/7.4	0.50/1.8	0.55/2.4	0.09/0.3
GBMV[e]	CHARMM	1.28/3.7	0.83/6.5	0.47/1.9	0.53/1.9	0.09/0.3
GBMV[f]	CHARMM	1.26/3.4	0.86/6.9	0.53/1.5	0.57/2.0	0.09/0.3
ACE22	CHARMM	12.0/43.0	25.3[g]/90.1	1.75/6.2	1.28/6.3	N/A
ACE22h	CHARMM	8.47/28.0	18.8[g]/110.2	2.45/8.0	1.81/5.1	N/A
ACE22n	CHARMM	12.4/44.5	25.9[g]/92.9	1.83/6.6	1.42/7.2	N/A
ACE22n[h]	CHARMM	8.52/28.8	19.5[g]/87.5	1.51/5.8	1.00/4.1	N/A
GBSW	CHARMM	5.21/23.7	9.87/65.1	1.41/6.8	1.05/5.7	0.25/0.6
GBSW	CHARMMSmooth	0.88/2.7	1.47/8.5	0.46/2.4	0.34/1.8	0.07/0.2
GBSW^{MS}	CHARMM	2.76/7.9	2.06/12.3	0.79/4.0	0.67/2.9	N/A
aGB^{HCT}	Amber94mBondi	1.25/3.2	2.27[a]/16.1	0.76/4.2	0.95/3.0	0.36/0.9
aGB^{OBC}(I)	Amber94Bondi	1.51/5.1	3.10/20.5	0.78/3.6	0.58/2.3	0.09/0.3
aGB^{OBC}(II)	Amber94Bondi	1.05/2.5	1.88/14.2	0.63/2.5	0.63/2.3	0.10/0.3
aGB^{JSB}	Amber94Eps2	7.69/18.1	13.50[a]/68.5	1.74/6.3	1.43/4.6	N/A
aGB^{MGB}	Amber94Eps2	3.90/9.3	5.05[a]/31.1	1.51/6.7	1.31/4.3	N/A
GBMV[i]	Amber94Bondi	1.11/3.0	1.28/5.0	0.41/1.5	0.38/1.5	0.10/0.3
GBMV[i]	Amber94Bondi	1.09/3.2	1.27/4.6	0.46/1.3	0.41/1.6	N/A
tGB^{HCT}	CHARMM	3.77/7.4	2.79[g]/15.9	1.23/7.0	1.06/4.2	N/A
tGB^{STILL}	CHARMM	3.88/9.2	3.87[g]/18.4	1.19/5.6	0.91/3.6	N/A
S-GB	OPLSImpact	1.66/4.1	2.10[g]/14.7	0.84/3.3	0.68/2.5	N/A

GB Method	RMSD (kcal/mol)					Shift (kcal/mol)				
	1	2	3	4	5	1	2	3	4	5
cGB^{STILL}	91.3	771.9[a]	33.4	45.5	274.5	−10.4	21.3[a]	61.6	116.1	455.4
GBMV[b]	14.5	32.3	3.7	8.1	11.5	−0.2	−7.3	7.1	−11.8	−29.9
GBMV[c]	14.1	25.4	3.2	7.4	12.6	6.9	0.1	11.8	0.7	−14.9
GBMV[d]	14.4	25.7	3.9	8.0	12.4	7.4	1.4	12.5	1.1	−14.5
GBMV[e]	15.2	25.4	3.9	7.9	12.6	7.3	1.2	12.2	0.1	−14.8
GBMV[f]	15.3	26.2	4.0	8.3	12.4	7.6	1.2	11.3	0.6	−13.9
ACE22	208.4	921.2[g]	12.6	19.5	N/A	−222.3	−332.1[g]	−96.8	−294.2	N/A
ACE22h	114.5	677.2[g]	23.6	25.5	N/A	−215.6	−374.1[g]	−186.1	−405.8	N/A
ACE22n	215.3	948.9[g]	13.1	21.8	N/A	−226.7	−337.8[i]	−100.5	−300.7	N/A
ACE22n[h]	138.8	729.3[g]	10.8	14.8	N/A	−173.5	−239.8[i]	−60.8	−216.2	N/A
GBSW	65.0	330.0	12.8	16.0	30.7	136.0	209.4	87.1	194.4	369.4
GBSW	13.9	50.7	4.4	6.0	9.1	−7.5	−19.4	10.3	2.4	−38.3
GBSW^{MS}	37.3	54.7	6.3	10.0	N/A	10.9	13.2	15.9	29.0	N/A
aGB^{HCT}	17.8	78.0[a]	7.2	14.7	50.7	−5.2	−1.5[a]	−13.5	−45.9	15.1
aGB^{OBC}(I)	25.4	112.0	6.9	9.3	12.3	34.9	56.7	22.5	34.4	104.6
aGB^{OBC}(II)	13.9	54.7	5.2	9.9	13.4	−13.2	−16.7	−16.3	−38.1	−68.2
aGB^{JSB}	118.7	505.2[a]	16.8	21.8	N/A	−167.2	−237.4[a]	−133.8	−256.2	N/A
aGB^{MGB}	50.2	177.1[a]	15.9	20.2	N/A	−48.9	−59.2[a]	−48.7	−130.3	N/A
GBMV[i]	15.3	41.8	3.3	6.1	13.6	−11.1	−18.7	−0.6	−24.3	20.3
GBMV[j]	15.0	41.2	3.7	6.5	N/A	−12.3	−18.8	−0.1	−23.5	N/A
tGB^{HCT}	48.7	76.7[g]	11.7	16.4	N/A	−13.1	−11.1[g]	−12.9	−30.7	N/A
tGB^{STILL}	45.7	115.7[g]	9.7	13.5	N/A	20.0	25.5[g]	28.0	48.9	N/A
S-GB	26.3	66.4[g]	8.4	11.8	N/A	−27.9	−56.9[g]	−33.2	−104.3	N/A

[a]Solvation energies could not be calculated for some structures due to negative Born radii.
[b]Original parameters and 38-point Lebedev grid.
[c]New parameters (see Methods) and regular angular grid with $N_\phi = 8$.
[d]New parameters and regular angular grid with $N_\phi = 5$.
[e]New parameters and 38-point Lebedev grid.
[f]New parameters and 26-point Lebedev grid.
[g]Solvation energies could not be calculated for some structures due to memory/program limitations.
[h]Atomic volumes scaled by 0.9.
[i]Amber 94 charges, Bondi radii,[90] new parameters, regular angular grid with $N_\phi = 8$.
[j]Amber 94 charges, Bondi radii, new parameters, regular angular grid with $N_\phi = 5$.
For each of the five test sets (1–5, see Table 1) the average and maximum error in solvation energies relative to the PB reference energies, the RMSDs of solvation energies, and the energy shift resulting in the smallest overall error are given.

reference energies are very small, partially as an artifact of the high charge resulting in very large numbers for the solvation energies. The energy RMS deviations are comparable with the other test sets.

Upon close examination the following general trends may be observed: First, the methods cGB^{STILL}, aGB^{JSB}, aGB^{MGB}, and all of the ACE versions do not provide solvation energies for proteins that are comparable in accuracy with the other methods. We do note, though, that aGB^{MGB} appears to be significantly improved over aGB^{JSB}, as suggested by the authors of these methods.[62] Further, ACE performs much better in the calculation of relative solvation energies between different conformations of villin and protein L, in test sets 3 and 4, than for relative solvation energies between different proteins in test sets 1 and 2. We also find that scaling the atomic volumes in ACE by 0.9 (as suggested by M. Schaefer, personal communication) does improve the results somewhat; however, they still do not reach the level of agreement with PB that is achieved with most GB methods.

The results from GBSW also carry quite substantial deviations. This is not too surprising because GBSW aims at modeling a somewhat different dielectric boundary definition.[14,87] In fact, if solvation energies from PB calculations with the same surface definition are used as the reference (CHARMMSmooth) the agreement is comparable to the best performance in Table 3. This would suggest that the deviations with respect to the PB energies for a molecular surface mostly reflect the extent to which solvation energies differ between van der Waals-based and molecular surface-based definitions. It was possible, however, to reparameterize GBSW to fit electrostatic solvation energies based on sharp molecular surface boundaries. Denoted as GBSW^{MS}, the deviations with respect to molecular surface PB energies then become much reduced.

Good agreement with PB solvation energies is in general found of the GBMV, aGB^{HCT}, aGB^{OBC}, and S-GB methods, where we find errors that are less than 3% for test set 2 and well below 1% for test sets 3 and 4. Excellent accuracy is found with the GBMV method in CHARMM, which achieves average errors of 1% or less for test set 2 and about 0.5% for different conformations of the same protein in sets 3 and 4 when the CHARMM force field is used. Using GBMV with the Amber force field and Bondi's radii[90] the error for test set 2 is slightly worse, at 1.3%, but better for sets 3 and 4. The most accurate method in Amber relative to the PB reference, aGB^{OBC}(II), is quite comparable to GBMV for test sets 3 and 4 but has somewhat larger errors for set 2. The accuracy of the S-GB method in Impact is similar but slightly worse compared to aGB^{OBC}(II) across all of the test sets.

The GB methods in Tinker, based directly on Still's formulation (tGB^{STILL})[51] and on the model by Hawkins, Cramer, and Truhlar (tGB^{HCT}),[56,89] also produce reasonably accurate solvation energies. The errors are comparable but in general slightly larger than the errors seen with very similar approaches in Amber.

The average relative errors and energy deviations compared to PB reference energies provide a good measure of the average performance of different GB methods in approximating the PB equation. For practical purposes it is equally important, however, to look at the maximum errors that one may encounter with a given method. For example, if one compares the GBMV results for the Amber force field and aGB^{OBC}(II) the average errors for structures in test set 2 are not that different, 1.28 and 1.88%, respectively. However, the maximum errors are only 5% with GBMV but 14.2% with aGB^{OBC}(II), which has consequences for the minimum level of accuracy that may be expected with either method for any given structure. In this respect, the S-GB method is also quite competitive with the GB methods in Amber with fairly low maximum errors despite somewhat larger average errors.

Table 4. Dependence of Errors in Solvation Energies from Selected GB Methods on Different PB Reference Energies.

		Average/maximum error (%)	
GB method	PB reference	2 (PDB)	3 (villin)
aGB^{OBC}(II)	PBEQ, 0.25 Å	1.88/14.2	0.63/2.5
aGB^{OBC}(II)	DelPhi, 0.5 Å	1.90/13.9	0.63/2.2
S-GB	PBEQ, 0.25 Å	2.10[a]/14.7	0.84/3.3
S-GB	DelPhi, 0.5 Å	2.07[a]/13.7	0.80/3.3

[a]Solvation energies could not be calculated for some structures, presumably due to size limitations.
Data are shown only for test sets 2 and 3.

We note that we are using PB reference energies that were calculated with the CHARMM PBEQ module, whereas many of the GB methods tested here used PB energies from DelPhi or other PB solvers were used as the reference for parameterization and testing. While the CHARMM PBEQ energies are very similar to energies obtained from other PB solvers (see below), one might suspect nevertheless that the agreement between GB and PB is somewhat better if the PB reference energies for the test cases are calculated in the same way as during the parameterization, e.g., with DelPhi. As shown in Table 4, we found that this does not appear to be the case. The data for test sets 2 and 3 suggest that the agreement is very much the same if compared to CHARMM PBEQ energies (0.25-Å grid spacing) or energies from DelPhi (0.5-Å grid spacing).

During our own efforts in developing and parameterizing GB methods we have noticed that the errors with respect to PB calculations are often systematic rather than random in nature. More specifically, we found that GB methods tend to have difficulties in reproducing relative solvation energies between folded and unfolded conformations and between smaller and larger protein structures. To investigate these points further we have taken a closer look at test sets 2 and 3.

Test set 3, with different conformations of the chicken villin headpiece generated from a lattice simulation protocol, really consists of three distinct subsets. The structures in the first subset come from short simulations that were started from extended conformations and for the most part have not yet reached compact configurations. The second subset contains structures that have been allowed to fold to compact protein-like, but misfolded, nonnative conformations. Structures in the third subset were simulated starting from the native conformation and are in general similar to the native fold or native-like. By comparing the average deviation in solvation energy for each of these subsets after shifting GB

Table 5. Average Deviation of Solvation Energies for Subsets of Test Set 3 as the Difference Between PB Reference and GB Energies after Substracting the Shifts Given in Table 3.

	Average deviation (PB-GB) (kcal/mol)		
GB method	1–40 (Extended)	41–80 (Misfolded)	81–120 (Native)
cGB^{STILL}	51.3	1.6	−14.6
GBMV[a]	1.7	0.7	−1.9
ACE22n[b]	−4.0	10.7	−6.5
GBSW	16.0	−4.0	2.1
$GBSW^{MS}$	4.1	2.3	−3.7
aGB^{HCT}	−5.7	1.1	1.1
aGB^{OBC}(I)	7.9	−2.4	0.8
aGB^{OBC}(II)	3.7	−1.7	1.4
aGB^{MGB}	−15.3	3.1	−1.5
tGB^{HCT}	3.2	2.7	−5.3
tGB^{STILL}	−9.5	3.3	0.0
S-GB	−7.3	2.2	0.0

[a]New parameters (see Methods) and regular angular grid with $N_\phi = 8$.
[b]Atomic volumes scaled by 0.9.
In this test set the first 40 villin headpiece structures were generated from short lattice simulations starting from extended chains, the second 40 compact (misfolded) structures that resulted from longer lattice folding simulations, and the third 40 structures originated from the native conformation (for annotations see Table 3).

energies according to the overall best shifts given in Table 3, systematic errors with respect to different types of conformations become apparent. Ideally, one would like to have average deviations of zero for all three subsets. Table 5 shows the actual deviations for a selected set of GB methods. Because the difference PB − GB is shown, positive values mean that conformations in a corresponding subset are too favorable relative to the other conformations and negative values mean that they are too unfavorable. This is dramatic, e.g., in cGB^{STILL}, where the solvation energies for extended structures are overestimated while the solvation energies for native-like conformations are not favorable enough, also relative to the misfolded conformations in the second subset. This would certainly pose a problem if such a method was used as part of a scoring function for finding the native structure according to the lowest energy in structure prediction efforts.[64] Some bias appears to be present in most GB methods and it is usually largest, positive or negative, for extended conformations. As one might expect, the more accurate methods also exhibit less of a bias in reproducing solvation energies for different types of conformations. The GBMV and aGB^{OBC}(II) methods also stand out here with deviations that are uniformly low for all subsets, suggesting that these methods would be in particular suited for applications such as structure prediction or protein folding, at least as judged by comparison to the PB treatment.

The issue of protein size-dependent errors is addressed in Table 6. Here, we have correlated the individual error in solvation energy for structures in set 2 with the radius of gyration and calculated the slope from a best-fit linear regression. Ideally, if there is no bias, the slope should be zero. Again, GBMV and aGB^{OBC}(II) meet this goal along with the other methods aGB^{HCT} and tGB^{HCT} and also $GBSW^{MS}$, the molecular surface-based reparameterization of the regular GBSW method. However, a significant systematic error with increasing protein size is found for the cGB^{STILL} and ACE and to a smaller degree also with aGB^{OBC}(I), aGB^{MGB}, and S-GB.

PB Methods

The deviations between solvation energies calculated by different PB methods were compared as well to provide reference points for

Table 6. Slope Calculated from Linear Regression of Individual Error (%) Versus Radius of Gyration for Structures in Test Set 2.

GB method	Slope in error (%) vs. radius of gyration
cGB^{STILL}	1.75[a]
GBMV[b]	0.07
ACE22n[c]	2.84[d]
GBSW	−1.61
$GBSW^{MS}$	−0.01
aGB^{HCT}	−0.07[a]
aGB^{OBC}(I)	−0.42
aGB^{OBC}(II)	0.05
aGB^{MGB}	0.57[a]
tGB^{HCT}	0.10[d]
tGB^{STILL}	0.16[d]
S-GB	0.24[d]

[a]Solvation energies could not be calculated for some structures due to negative Born radii.
[b]New parameters (see Methods) and regular angular grid with $N_\phi = 8$.
[c]Atomic volumes scaled by 0.9.
[d]Solvation energies could not be calculated for some structures due to memory/program limitations.

evaluating the performance of GB methods. We were in particular interested in comparing faster PB methods that provide moderate accuracy at low computational cost because such methods would be in direct competition with GB methods for the calculation of solvation energies.

The results from comparing PB methods are summarized in Table 7. First, one may notice that PB methods that would be expected to be of similar accuracy as the reference PBEQ calculations do in fact give very similar results for all of the test sets, with deviations of 0.15–0.3%. These methods are MEAD with a 0.25-Å grid spacing, APBS with a 0.4-Å grid spacing (with the molecular surface boundary), and DelPhi with a 0.5-Å grid spacing. The observation that APBS and DelPhi match the 0.25-Å grid PBEQ energies with coarser grid sizes translates into better computational efficiency, as discussed in more detail below. As one would expect, the accuracy is reduced with all methods if coarser grid spacings are used. With DelPhi, it is still possible, however, to obtain fairly low errors of around 0.5%, even when a rather coarse 1.0-Å grid spacing is used. This may be compared with the best GB methods that show deviations between 0.5 and 1.0% depending on the test set. The errors found in the best GB methods are otherwise comparable to PBEQ and MEAD results with 0.5-Å grid spacing and also to the level of accuracy achieved with the fast boundary element method REBEL.

As one would expect, different surface definitions do in fact result in quite different solvation energies. This is the case for APBS with a harmonically averaged molecular surface and especially with the spline-interpolated van der Waals-based smooth boundary definition where the deviations are quite significant. When compared to solvation energies obtained with CHARMM PBEQ for the same kind of surface, the deviations are reduced but remain large overall. This would suggest other differences in the boundary definition because the solvation energies agree well between PBEQ and APBS for the standard molecular surface with the same set of APBS parameters. The solvation energies obtained with the finite element method PBF in Impact also show significant deviations from the finite element solutions with PBEQ or DelPhi, as would be expected from the different surface definition based on the smooth dielectric boundary as proposed by Gilson et al.[94] However, the solvation energies do agree fairly well with energies obtained by APBS with the harmonically averaged surface when recalculated with the OPLS force field and radii to match the Impact input parameters (data not shown), suggesting a very similar surface definition.

Finally, ZAP, which uses Gaussian functions to approximate the molecular surface, produces solvation energies with deviations that match average GB methods but do not quite reach the level of agreement with PB reference energies that is found with the best GB methods.

Timing Data

Because GB methods are meant to approximate PB solutions at a much reduced cost, a performance evaluation needs to consider the computational expense necessary to achieve a certain level of accuracy. For obtaining timing information on GB calculations we considered two cases: The first is a single calculation of the solvation energy for a given structure, which is directly comparable to PB calculations; the second measures the time required to calculate gradients in a 10-step steepest descent minimization run, which is also relevant for running molecular dynamics simulations. Two test systems were used: The first, 1VII, is the 36-residue chicken villin headpiece; the second, 1DVJ_A, is chain A from oritidine monophosphate decarboxylase, which is much larger with 239 residues. The timing data for GB calculations are given in Table 8. We find that all of the methods implementing a pairwise sum for the calculation of Born radii—such as cGB^{STILL}, aGB^{HCT}, aGB^{OBC}(I), aGB^{OBC}(II), aGB^{JSB}, aGB^{MGB}, tGB^{HCT}, and tGB^{STILL}—require approximately similar amounts of time. However, it was possible to gain almost a factor of two with newly optimized routines in a prerelease version of Amber 8 for the aGB^{HCT}, aGB^{OBC}(I), aGB^{OBC}(II), and aGB^{JSB} models. We note that the timings are platform dependent and the results may differ somewhat on other computer architectures. The more accurate GBMV method based on an analytic integration of the molecular volume is in general slower, although the computational expense scales more favorably with increasing system size if compared to the current implementation in Amber. If no cutoffs are used, which is most relevant for the calculation of complete solvation energies, the timings for the larger structure 1DVJ_A are actually quite competitive with the other GB methods while the calculations are about three to four times more expensive for the smaller structure, 1VII. In dynamics simulations, interaction cutoffs are commonly used. Here, the pairwise methods gain a significant advantage over the volume integration scheme where the computational cost is not reduced as much as a function of cutoff. In the case of the larger system this translates into a factor of four to five for dynamics with the GBMV method versus the newly optimized pairwise implementation in Amber 8. Finally, we note that the computational times for S-GB are significantly larger than for the other GB methods, both for the small and the large test structure.

The time to calculate single energies of solvation with GB can be compared to the time required for different PB methods as shown in Table 9. Timings for grid-based PB methods depend foremost on the number of grid points, given similar convergence criteria. Therefore, constraints on what kinds of grids can be used with a given algorithm (rectangular as in PBEQ, cubic as in MEAD and APBS, or limited choice of grid dimensions as multiples of 32 in APBS) strongly influence the timing, and a simple comparison between methods based on the data given in Table 9 is difficult. Nevertheless, APBS appears to be more expensive per grid point, presumably due to extra expense in the multigridding finite element approach, but (as shown above) a coarser grid resolution appears to be sufficient in APBS to obtain a similar level of accuracy as the standard finite difference method PBEQ, which makes APBS overall more efficient. Another issue that arises when comparing PB methods in terms of efficiency is the choice of convergence criteria. This is most apparent with the MEAD program. With the default settings in the latest version, fairly tight convergence criteria are used with increasing grid size to maintain the same level of long-range accuracy. As a result, the time required for reaching convergence increases much faster for larger grid sizes than with the comparable PBEQ module in CHARMM. With more relaxed convergence criteria (using the flag –convergence_oldway), the resulting solvation energies are quite similar

Table 7. Comparison of Electrostatic Solvation Energies from PB Methods with Appropriate PB Reference Energies.

PB method	Grid (Å)	PB reference	Average/maximum deviation (%) Set 1 (training)	Set 2 (PDB)	Set 3 (villin)	Set 4 (protein L)	Set 5 (myloglobin)
PBEQ	0.40	CHARMM	0.52/2.7	0.71/7.0	0.34/1.2	0.22/0.7	0.10/0.3
PBEQ	0.50	CHARMM	0.75/2.4	1.25/12.5	0.60/1.7	0.40/1.4	0.13/0.4
MEAD	0.25	CHARMM	0.21/0.9	N/A	0.16/0.8	N/A	N/A
MEAD	0.50	CHARMM	0.98/3.4	N/A	0.62/2.2	N/A	N/A
DelPhi	0.50	CHARMM	0.16/0.4	0.23/2.7	0.18/0.7	0.14/0.5	0.08/0.2
DelPhi	1.00	CHARMM	0.50/2.2	0.59/4.5	0.56/1.6	0.37/1.5	0.10/0.3
APBS[a]	0.40	CHARMM	0.29/0.8	0.23[b]/2.6	0.31/1.2	0.27/1.1	0.13[b]/2.3
APBS[a]	0.50	CHARMM	0.63/2.4	0.61/5.2	0.51/1.5	0.36/1.4	0.19/1.8
APBS[c]	0.40	CHARMM	1.41/6.8	2.44[b]/21.1	0.70/2.3	0.41/1.4	N/A
APBS[c]	0.50	CHARMM	1.51/6.7	2.66/23.2	0.74/2.3	0.44/1.3	N/A
APBS[d]	0.40	CHARMM	16.1/73.6	N/A	3.95/14.8	1.36/4.8	N/A
APBS[d]	0.50	CHARMM	16.9/73.1	N/A	4.27/12.0	1.52/5.3	N/A
APBS[d]	0.40	CHARMMsmooth	10.0/33.8	N/A	2.34/7.0	0.72/2.9	N/A
APBS[d]	0.50	CHARMMsmooth	10.7/39.1	N/A	2.65/6.9	0.90/3.5	N/A
PBF	Low[e]	OPLSImpact	1.33/5.7	3.28[b]/31.2	0.89/2.8	0.73/3.1	N/A
PBF	High[e]	OPLSImpact	1.64/8.8	3.95[b]/33.3	1.06/3.4	0.61/2.1	N/A
ZAP	0.25	CHARMM	1.11/5.1	2.63/18.9	0.72/3.9	0.90/3.9	2.11/3.9
ZAP	0.50	CHARMM	1.10/5.0	2.47/17.7	0.72/3.7	0.90/3.9	0.31/0.7
REBEL	Default	Amber94Bondi	1.33/5.1	1.29/10.5	0.37/1.2	0.46/1.9	N/A
REBEL	Exact	Amber94Bondi	0.48/2.0	N/A	0.21/0.9	0.23/0.9	N/A

PB method	Grid (Å)	RMSD (kcal/mol) 1	2	3	4	5	Shift (kcal/mol) 1	2	3	4	5
PBEQ	0.40	5.8	20.7	3.0	3.3	12.7	13.8	21.6	10.5	24.6	64.9
PBEQ	0.50	10.4	36.5	5.1	6.0	16.5	23.3	38.0	19.1	42.3	103.0
MEAD	0.25	4.0	N/A	1.2	N/A	N/A	−3.8	N/A	−3.2	N/A	N/A
MEAD	0.50	13.1	N/A	4.9	N/A	N/A	22.0	N/A	19.4	N/A	N/A
DelPhi	0.50	3.4	8.4	1.4	2.1	9.8	1.1	2.7	−0.8	0.5	44.0
DelPi	1.00	6.6	19.0	4.3	5.7	12.8	6.0	12.6	4.0	11.4	53.6
APBS[a]	0.40	3.6	27.3[b]	2.4	4.1	37.1	3.1	3.4	3.5	4.5	5.6
APBS[a]	0.50	7.7	18.0	4.2	5.2	34.5	11.0	18.9	11.9	21.6	56.2
APBS[c]	0.40	17.4	71.5[b]	6.8	5.9	37.1	−42.5	−70.0	−34.7	−84.3	N/A
APBS[c]	0.50	18.9	79.4	7.1	6.4	29.3	−45.8	−76.9	−38.0	−90.9	−193.3
APBS[d]	0.40	215.6	N/A	35.2	20.3	N/A	552.2	N/A	349.7	742.0	N/A
APBS[c]	0.50	226.2	N/A	36.4	22.8	N/A	571.1	N/A	368.4	777.5	N/A
APBS[c]	0.40	162.0	N/A	24.7	12.3	N/A	384.8	N/A	274.2	550.7	N/A
APBSc	0.50	166.2	N/A	26.1	15.6	N/A	424.6	N/A	293.6	584.4	N/A
PBF	Low[e]	21.3	102.6[b]	8.3	13.0	N/A	−40.1	−87.3	−43.5	−80.0	N/A
PBF	High[e]	24.2	121.8[b]	10.0	11.0	N/A	−60.1	−112.2	−55.4	−110.8	N/A
ZAP	0.25	17.8	98.3	7.0	14.1	297.9	7.5	25.3	−2.6	5.1	−352.3
ZAP	0.50	17.7	93.8	7.0	14.0	42.1	6.9	25.8	−1.7	5.4	−1.0
REBEL	Default	33.6	58.3	3.0	7.2	N/A	8.9	10.9	1.3	−6.9	N/A
REBEL	Exact	7.3	N/A	1.7	3.5	N/A	9.5	N/A	5.1	14.2	N/A

[a]Molecular surface.
[b]Solvation energies could not be calculated for some structures, presumably due to size limitations.
[c]Molecular surface with simple harmonic averaging.
[d]Spline-interpolated van der Waals-based smooth dielectric boundary.
[e]Low and high resolution according to default settings in Impact program.
For each of the five tests (1–5, see Table 1) the average and maximum error in solvation energies relative to the PB reference energies, the RMSDs of solvation energies, and the energy shift resulting in the smallest overall error are given.

Table 8. Timing of Single Calculations of Electrostatic Solvation Energies with GB Methods and 10-Step Steepest Descent Minimizations on Pentium IV (2 GHz) with Standard Distributions and/or Default Compiler Flags for 1VII (36 Residues, 596 Atoms) and 1DVJ_A (239 Residues, 3628 Atoms).

	Time (s) for single energy calculation		Timing (s) for 10-steps minimiation			
			1VII		1DVJ_A	
GB method	1VII (no cutoff)	1DVJ_A (no cutoff)	(No cutoff)	14-Å cutoff)	No cutoff	14-Å cutoff
cGB^{STILL}	0.2	4.7	3.1	2.6	68.5	16.6
GBMV[a]	1.3	9.5	14.4	14.1	116.8	90.0
GBMV[b]	0.6	5.1	7.2	6.8	69.4	43.0
GBMV[c]	0.7	5.8	8.1	7.8	75.4	49.6
GBMV[d]	0.6	4.7	6.3	6.0	63.8	37.3
ACE	0.2	4.9	3.1	2.6	69.7	17.1
GBSW	0.3	3.8	3.7	3.4	49.6	23.3
$GBSW^{MS}$	0.4	4.3	4.6	4.3	53.8	27.4
aGB^{HCT}	0.2/0.1	5.7/2.6	2.6/1.9	2.1/1.5	69.1/39.0	18.6/11.7
$aGB^{OBC/JSB}$	0.2/0.1	5.5/2.5	2.6/1.9	2.1/1.5	69.8/39.0	19.5/11.7
aGB^{MGB}	0.2	6.5	2.8	2.3	78.6	20.3
tGB^{HCT}	0.2	5.1	N/A	N/A	N/A	N/A
tGB^{STILL}	0.2	7.2	N/A	N/A	N/A	N/A
S-GB	1.9	56.0	N/A	N/A	N/A	N/A

[a]New parameters (see Methods) and regular angular grid with $N_\phi = 8$.
[b]New parameters and regular angular grid with $N_\phi = 5$.
[c]New parameters and 38-point Lebedev grid.
[d]New parameters and 26-point Lebedev grid.
Single energy calculations do not include overhead that is unrelated to GB. Steepest descent timings measure the complete time including the time required for setup and calculation of non-GB energy terms. The two numbers given for the GB methods implemented in Amber refer to Amber 7 and a prerelease version of Amber 8.

but the time necessary to calculate solvation energies is reduced substantially to about the same level as with CHARMM PBEQ.

If compared to PBEQ, APBS, and MEAD we find that DelPhi requires about the same amount of time if the same grid spacing, 0.5 Å, is used with all of these methods. It needs to be taken into account, however, that the accuracy is comparable to PBEQ with 0.25 Å and slightly better than APBS with a 0.4-Å grid spacing, which translates into an effective speedup of 10 (1VII) or 5 (1DVJ_A) versus the PBEQ method. While this is impressive compared to other PB solvers, it still leaves enough room for GB methods where the timings with some methods are better by at least an order of magnitude even if a 1-Å grid spacing is used with DelPhi. The boundary element method REBEL comes much closer to the speed needed for GB calculations. Slower by less than a factor of two for larger molecules and about the same speed as GBMV for the small villin headpiece, it provides a similar level of accuracy as GBMV and is thus quite competitive with GB methods in terms of speed and accuracy for the calculation of solvation energies.

Finally, ZAP clearly fulfills its promise of being a very fast PB solver. The times are much shorter than with all of the other PB solvers and, for the larger molecule 1DVJ_A, also shorter than most GB methods (see Table 8).

Overall Performance

The results discussed so far are summarized best in plots of accuracy versus time as given in Figures 1 and 2, which compare the time required for 1VII solvation energies against the error in test set 3 (Fig. 1) and the time required for 1DVJ_A against the error in test set 2 (Fig. 2). Both figures are shown on a double logarithmic scale to represent the extended range of time and accuracy for all of the methods tested here. While the accuracy for the reference PB calculations with PBEQ and a grid spacing of 0.25 Å are technically 0 there is certainly an error associated with these PBEQ calculations with respect to a hypothetical set of fully converged reference energies at infinitely fine grid spacing. Because both MEAD and PBEQ essentially implement the same accurate, finite difference method, the deviations between solvation energies from both methods at 0.25-Å grid spacing may be seen not just as an error estimate for MEAD but also for PBEQ. Accordingly, we would estimate that PBEQ at 0.25-Å grid spacing has an error of 0.2% for test set 2 (from the accuracy of MEAD for test set 1) and 0.16% for test set 3. Another measure of the error associated with finite difference PB solutions may be obtained from the variation of solvation energies upon rotation or translation of a given molecule. While this error may be as large as a few percent for a single ion,[19] it is much smaller for polyionic molec-

Table 9. Timing of Single Calculations of Electrostatic Solvation Energies with PB Methods on Pentium IV (2 GHz) with Standard Distributions and/or Default Compiler Flags for 1VII (36 Residues, 596 Atoms) and 1DVJ_A (239 Residues, 3628 Atoms).

	Timing (s) for single energy calculation	
PB method	1VII	1DVJ_A
PBEQ, 0.25 Å	**65.3**	**529.4**
	151 × 135 × 115	*254 × 241 × 195*
PBEQ, 0.40 Å	**14.0**	**108.2**
	95 × 85 × 73	*157 × 151 × 121*
PBEQ, 0.50 Å	**7.6**	**54.1**
	75 × 67 × 57	*127 × 121 × 97*
DelPhi, 0.50 Å	**6.5**	**139.3**
DelPhi, 1.00 Å	**1.5**	**33.5**
APBS, 0.40 Å	**30.8**	**227.2**
	97 × 97 × 97	*161 × 161 × 161*
APBS, 0.50 Å	**13.2**	**122.9**
	65 × 65 × 65	*129 × 129 × 129*
MEAD, 0.25 Å	**105.9/54.9**	**2377.0/993.9**
	121 × 121 × 121	*231*
MEAD, 0.50 Å	**8.7/7.3**	**154.5/109.9**
	65 × 65 × 65	*115 × 115 × 115*
ZAP, 0.50 Å	**0.3**	**2.9**
PBF, low resolution	**22.0**	**291.3**
PBF, high resolution	**83.3**	**1018.0**
REBEL, default	**0.8**	**10.9**
REBEL, exact	**8.1**	**N/A**

The two numbers given for MEAD refer to new and old convergence criteria, respectively. For DelPhi convergence criteria of 0.1 kT/e were requested. Default convergence criteria were used in all other cases. Finite difference grid dimensions are given in italics.

ular systems due to cancellation of errors. For a small molecule (52 residues) from test set 1 (1MBG) we found a standard deviation of 0.62 kcal/mol when solvation energies are calculated for different molecular orientations with the PBEQ method using a grid spacing of 0.25 Å. This corresponds to a relative error of 0.05% with respect to the total solvation energy.

Figures 1 and 2 demonstrate how different methods trade speed for accuracy covering the range from fast and approximate to slow and accurate ways of calculating solvation energies. It can be seen how some methods—such as the GB methods aGB^{OBC}(II) and GBMV and the PB solvers REBEL and DelPhi—are pushing the envelope in different regimes of accuracy and speed, but the lack of data points in the lower left corners of both figures also suggest fundamental limits of what can be achieved in terms of fast *and* accurate calculations of electrostatic solvation energies with GB or PB methods.

Finally, we would like to focus on the performance of GB methods when used for molecular dynamics simulations. For this case the minimization timings with typical 14-Å nonbonded cutoffs as given in Table 8 are most relevant and may be compared with the errors obtained for test sets 3 and 4. For the purposes of this comparison we would consider the somewhat larger errors in test set 2 less representative because they may indicate systematic

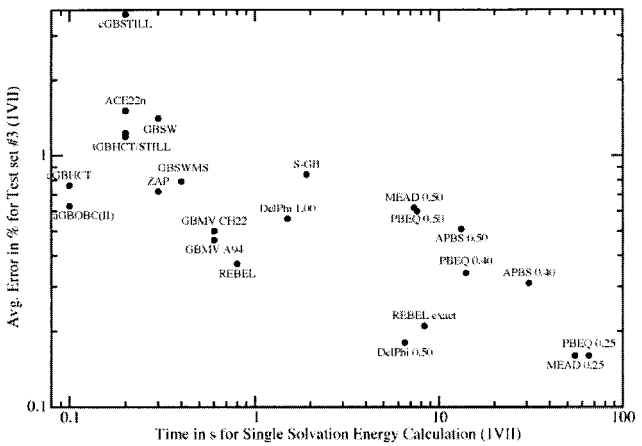

Figure 1. Performance of GB and PB methods as measured by average percentage error for test set 3 (1VII conformations) versus the time required for a single calculation of the electrostatic solvation energy for the native structure 1VII. Both time and error axes are shown logarithmically. The labels indicate the source of data points shown.

errors due to different amino acid compositions that would cancel out to some extent in the relative energies between different conformations of the same protein in simulations. Figures 3 and 4 show such comparisons for 1VII and 1DVJ_A with selected GB methods in the CHARMM and Amber programs. Again, one can see the trade-off between speed and accuracy from aGB^{OBC}(II) to the most expensive GBMV method with a high integration grid (N_ϕ = 8). One may note that the GBMV methods become relatively more expensive when cutoffs are used because the grid-based integration method for calculating Born radii always considers the solute volume up to a cutoff of 18 Å. With GBMV, the introduction of cutoffs only reduces the time spent for the appli-

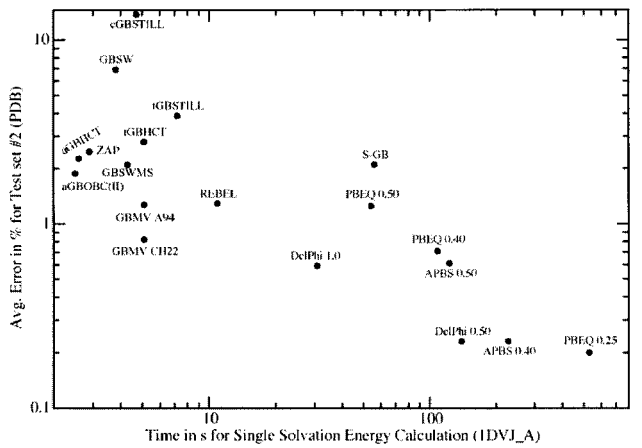

Figure 2. Performance of GB and PB methods as measured by average percentage error for test set 2 (PDB conformations) versus the time required for a single calculation of the electrostatic solvation energy for the native structure 1DVJ_A. Both time and error axes are shown logarithmically. The labels indicate the source of data points shown.

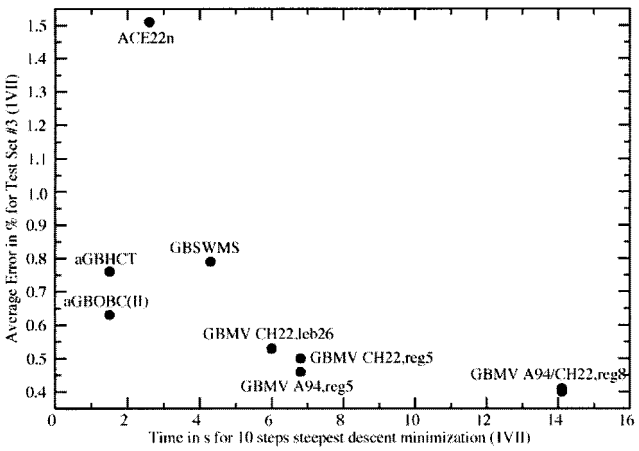

Figure 3. Performance of selected GB methods in CHARMM and Amber as measured by average percentage error for test set 3 (1VII conformations) versus the time required for 10 steps of steepest descent minimization for 1VII. The labels indicate the source of data points shown. Different integration grids were used for GBMV and are annotated as follows: leb26, 26-point Lebedev grid; reg5, regular angular grid with $N_\phi = 5$; reg8, regular angular grid with $N_\phi = 8$.

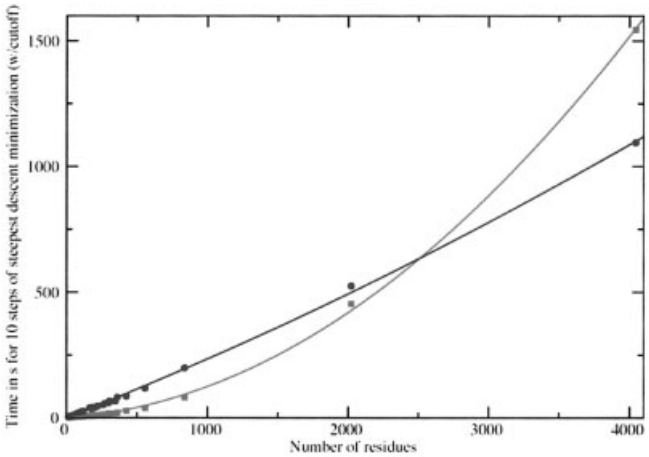

Figure 5. Time required with aGB^{OBC}(II) (squares) and GBMV with $N_\phi = 5$ (circles) for 10 steps of steepest descent minimization versus number of residues for selected protein structures.

cation of the generalized Born equation [eq. (6)] and of course the time spent for the calculation of intrasolute, nonbonded interactions that are unrelated to GB.

The GBMV method as implemented in CHARMM does scale better, however, with an increase in system size because the speedup of aGB^{OBC}(II) versus GBMV ($N_\phi = 5$) is reduced from a factor of 4.2 for the 36-residue protein1VII to 3.7 for the 239-residue protein 1DVJ_A. We have investigated this point further to estimate the performance for very large systems and compared the timing with aGB^{OBC}(II) and GBMV ($N_\phi = 5$) for a number of different proteins from test 2 as well as the dimer and tetramer of β-galactosidase from *Escherichia coli* (PDB code 1DP0), which has 2022 residues for the dimer and 4044 residues for the biologic tetramer unit. The results are shown in Figure 5, illustrating the different scaling behavior depending on system size, as measured by the number of residues. As is evident, the GBMV method in CHARMM follows nearly linear scaling with the number of residues, N, while the pairwise aGB^{OBC}(II) method as currently implemented in Amber is dominated by O(N^2) scaling. As a consequence there is a crossover point around 2500 residues for a cutoff of 14 Å. For shorter cutoffs this crossover point would move to a larger number of residues and for larger cutoffs to a smaller number of residues.

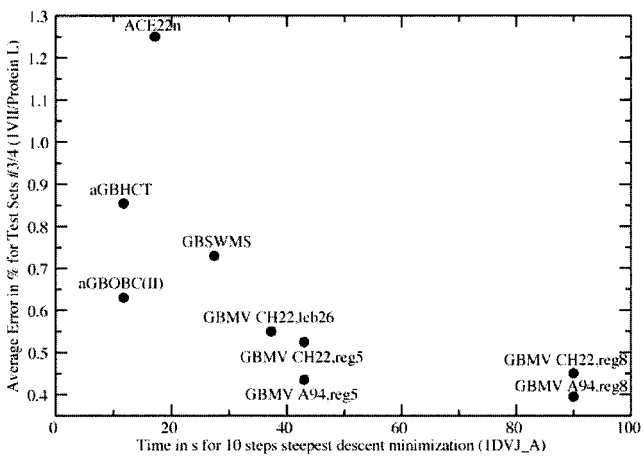

Figure 4. Performance of selected GB methods in CHARMM and Amber as measured by average percentage error for test sets 3 (1VII conformations) and 4 (protein L conformations) versus the time required for 10 steps of steepest descent minimization for 1DVJ_A. The labels indicate the source of data points shown as in Figure 3.

Discussion

In this study we have compared a variety of GB methods and PB solvers, some of which were developed in our own laboratories, in terms of accuracy and speed for the calculation of electrostatic solvation energies. The present study is focused on protein structures, but the results are expected to apply at least qualitatively to other macromolecular systems of similar size and composition as well. As the point of reference for such calculations, high-resolution solutions to the Poisson equation, obtained by finite difference methods, for example, present the natural choice because both the GB and PB models share the same physical basis. The agreement with experimental data or explicit solvent simulations is not being addressed here. To the extent that GB methods provide good approximations to the "correct" solvation energies as described by the Poisson equation they also share the successes and failures due to such a continuum solvent environment. Other studies have described the agreement of PB calculations with experimental transfer energies from vacuum to water[97] or with experimental pK_a values.[30] Further evaluation of implicit solvent models, especially based on dynamic trajectories in simulations that are now becom-

ing possible with GB and some PB methods, is ongoing in our laboratories.[81]

The comparisons here assume a simple two-dielectric model with the dielectric boundary defined by the molecular surface as the most appropriate target for GB and PB methods. This choice of reference is motivated in particular by the fact that most GB and PB methods are parameterized based on the molecular surface and should therefore be compared to such. However, it then becomes problematic when comparing to methods such as GBSW,[87] which explicitly target alternative surface definitions.

When comparing different GB methods in terms of accuracy we find a continuous range from earlier GB models, cGB^{STILL} and tGB^{STILL}, which are based on a pairwise approximation for calculating Born radii, to a variety of newer implementations, either based on the pairwise descreening approach by Hawkins, Cramer, and Truhlar[56] in aGB^{HCT}, aGB^{OBC}(I), and aGB^{OBC}(II) as well as a new surface integral approach[59] in S-GB. The highest accuracy was achieved with a volume integral formalism and additional correction to the Coulomb field approximation in GBMV and also GBSW. However, the better agreement with PB reference energy in GBMV does come at the price of computational cost. As Tables 3 and 8 as well as Figures 3 and 4 show, the performance of GBMV can be tuned to some degree between higher accuracy/slower execution time and less accuracy/faster execution. There is also some variation depending on the parameterization and the underlying force field, as the error for test set 2 varies from 0.8% with parameters optimized for CHARMM22 using test set 2 to almost 1.3% with the same parameters using the Amber force field. Previous parameterizations of GBMV based on smaller training sets were found to produce similar errors within the same range. We recognize that the error estimate of 0.8% for the CHARMM22 force field may be slightly too optimistic because we are reporting the error for the training set in this case rather than an independent test set. We think that this number is nevertheless relevant because it gives a lower error estimate for the GBMV method when calculating solvation energies for different protein structures, but the test set is also so comprehensive in terms of size, fold topologies, and amino acid composition that it is difficult to imagine that truly independent test sets can be found that would represent qualitatively different types of native structures and possibly result in significantly larger errors. However, to be on the safe side, one may use the error of 1% for test set 2 from the previous parameterization as a conservative error estimate for GBMV.

Results from the GBSW method agree equally well with solvation energies from PB if the same smooth surface definition is used. If GBSW is reparameterized to reproduce molecular surface PB solvation energies the agreement is still good, as the data for $GBSW^{MS}$ shows, while the calculation based on overlapping van der Waals spheres is computationally more efficient than the explicit calculation of the molecular surface in GBMV.

The best method based on a pairwise calculation of Born radii, aGB^{OBC}(II), is faster by a factor of 2–6 depending on the size of the protein. At this point, the choice of either method becomes a question of accuracy and speed requirements for a certain application. For example, somewhat lower accuracy may be acceptable for scoring large sets of conformations or the generation of conformational ensembles with molecular dynamics simulations while speed would certainly be of the essence there. On the other hand, refinement of near-native structures or the calculation of binding energies may require the best possible level of accuracy if even that would entail higher computational costs.

While the overall error in calculating solvation energies with GB methods should, obviously, be as small as possible, it is even more important that systematic errors are avoided. We looked at two aspects in this regard: systematic errors with increasing system size and systematic errors for extended conformations versus compact native-like and misfolded conformations. We found that only the most accurate methods, aGB^{OBC}(II) and GBMV, are mostly free of systematic biases in either case, while in some cases solvation energies for larger structures or extended structures are significantly over- or underestimated. This also means that some of the less accurate methods may be quite reasonable for a limited subset of conformations, such as small native-like structures, while lacking the general applicability to other structures. This is not be too surprising because most of the GB methods tested here were parameterized with small molecules and/or smaller protein structures and our study did not assess in detail whether it would be possible to reparameterize underperforming methods to improve the results.

Finally, it should be mentioned that the methods cGB^{STILL} and ACE have been more successful with the united-atom force field CHARMM19,[55,63,68,98–100] which was not included in the present study.

It is important to put the performance of GB methods in the context of more approximate PB methods to verify that even the slower and more accurate GB methods such as GBMV still provide an advantage for calculating solvation energies in terms of speed. We find that all of the PB solvers that we have tested, except for ZAP and REBEL, do in fact require much more time for the same level of accuracy achieved with the best GB methods. ZAP and REBEL stand out on their own by providing fast solvation energies with an accuracy and speed that are comparable to the performance of GB methods. We find that ZAP is very fast but slightly less accurate than aGB^{OBC}(II) and GBMV and REBEL is as accurate as GBMV but noticeably slower.

A comparison of PB methods based on the data presented here is not comprehensive because many details of PB solvers that influence speed and accuracy were not tested exhaustively, such as issues of convergence and grid size. Nevertheless, we think that we can identify certain trends. The finite difference and finite element PB solvers CHARMM PBEQ, MEAD, DelPhi, and APBS appear to be all fairly equivalent in terms of accuracy, with relative errors of about 0.2%. A more detailed comparison at this level of agreement would require a better set of reference energies as well as a comparison with analytic results for simple geometries. The observed differences in speed on a larger scale can be explained mostly by the grid resolutions required in different methods for obtaining the same level of accuracy. Both APBS and DelPhi are faster than PBEQ and MEAD because grid spacings of 0.4 and 0.5 Å in APBS and DelPhi, respectively, provide the same accuracy as grid spacings of 0.25 Å in PBEQ and MEAD. In DelPhi, this is achieved by interpolating the location of surface charges that are used to calculate the reaction field. APBS corrects solvation energies from coarser grid resolutions based on an error estimate to match solvation energies from higher grid resolutions. DelPhi

has further advantages in terms of speed by using only a single finite difference solution for calculating the reaction field directly rather than the two passes that are required in the other methods for calculating the difference according to eq. (4). Larger grid spacings for a given method in general reduce the computational time significantly, but the trade-off in accuracy is not competitive with most of the GB methods.

Another important aspect when comparing PB solvers is the treatment of salt. We did not take salt effects into account in this study because salt dependence is not implemented in many of the GB methods tested here. The inclusion of salt requires solution of the Poisson–Boltzmann rather than the Poisson equation, either in its original nonlinear or an approximate linearized form. We expect that this would change the relative performance of different PB solvers significantly and we should stress that the results from this study are certainly not applicable in the same way to calculations that take salt into account.

Finally, we have looked in more detail at the performance of the best GB methods, aGB^{OBC}(II) and GBMV, for minimization and molecular dynamics applications. The computational cost of these methods depends differently on the choice of an electrostatic cutoff and the size of the system. The pairwise formulation in aGB^{OBC} scales much better with the introduction of cutoffs than the integral formulation in GBMV, resulting in a difference of speed by a factor of 3–4 for smaller integration grid sizes. It remains to be seen whether the better accuracy with GBMV justifies the extra expense, especially if one keeps in mind that even an ideal PB model presents only a fairly crude approximation to the intricate details of explicit solvent interactions. We find, though, that the pairwise formulation currently implemented in Amber scales more like $O(N^2)$ with the number of residues while the GBMV implementation in CHARMM scales closer to O(N) if a cutoff is used. A difference in scaling is explained in part by the different methods for calculating Born radii, but the dominant $O(N^2)$ term in the aGB methods is a consequence of not using a nonbonded pair list in the Amber implementation. While this improves performance for small systems it increases computational expenses for large systems. This observation is noteworthy because the ability to efficiently model large protein multimers and supramolecular assemblies will become increasingly important in the future.

It will be interesting to see whether new developments in the area of PB-based continuum solvent descriptions will come from improving PB-based, GB-based, or other novel approximate methods. While highly accurate Born radii can now be calculated with GB methods such as GBMV, limitations in the generalized Born formalism as given by eq. (6) are becoming more obvious as the errors in calculating solvation energies approach those that can be obtained with "perfect" Born radii from PB solutions.[50] Future improvements in terms of accuracy for GB methods will likely have to focus on finding a better expression than eq. (8) for representing the cross-polarization energies for a polyionic system. More approximate PB methods, on the other hand, could benefit from new developments for accurate representation of molecular surfaces through analytic functions that have been developed for GB methods[60] but could be applied in PB solvers such as Zap as well.

Conclusions

In this study we have attempted to provide a comprehensive test of GB and PB methods for the calculation of electrostatic solvation energies. We found a wide range of accuracy and speed that is spanned from fast and fairly accurate newly parameterized GB methods to PB solvers with high accuracy but significant computational cost. The latest GB methods can achieve errors of approximately 1% for solvation energies of different protein structures and 0.4% for relative energies of different conformations for a single structure, which is a significant improvement over first implementations of the GB methodology. However, higher levels of accuracy can only be reached with much more expensive finite difference or finite element PB methods at this point. Fast, alternative methods for solving the PB equation based on boundary element methods, as in REBEL, or Gaussian volume overlap in Zap were found to be competitive with GB methods, although still either slower or less accurate, respectively, than the best GB methods. Here, we are focusing mainly on the calculation of electrostatic solvation energies for single conformations. An important advantage of most GB methods is the availability of analytic derivatives for use in molecular dynamics simulations or gradient-based energy minimization. Obtaining smooth derivatives from PB methods that provide suitable forces for simulations is often difficult or impossible. This point should be taken into consideration when fast PB solvers such as REBEL are compared to GB methods with a similar level of performance.

The results presented here are relevant to the application of continuum models in a variety of applications such as the scoring of protein conformations[64] or the estimation of conformational free energies[34] by giving an estimate of what kind of performance can be expected with different GB or PB methods. The much improved level of accuracy in analytic GB methods, which are in particular well suited for simulation studies, also provides an encouraging basis for more reliable implicit solvent simulations and opens a whole new range of possibilities for reaching larger system sizes and longer timescales.

Acknowledgments

The authors acknowledge Prof. Donald Bashford for helpful suggestions and discussions. They thank Dr. Michael Schaefer for help with ACE and a set of the most recent parameters. They are grateful to Schrödinger, Inc., for providing the FirstDiscovery suite, Openeye Software for providing ZAP, and MolSoft, LLC, for providing the ICM program. They also thank Dr. Maxim Totrov from MolSoft for comments and suggestions with respect to ICM/REBEL and technical support from Schrödinger, Inc., for comments with respect to Impact/PBF. Financial support from the National Institutes of Health (NIH)-supported resource Multiscale Modeling Tools in Structural Biology (http://mmtsb.scripps.edu) (NIH Grant RR12255) and from NIH Grant GM57513 is acknowledged.

Appendix

PDB Codes, Test Set 1

1AJJ, 1BBL, 1BOR, 1BPI, 1CBN, 1FCA, 1FRD, 1FXD, 1HPT, 1MBG, 1NEQ, 1PTQ, 1R69, 1SH1, 1SVR, 1TSG, 1UXC, 1VII, 1VJW, 2ERL, 2PDE, 451C.

PDB Codes, Test Set 2

1A23, 1A2S, 1A5R, 1A63, 1A66_A, 1A6B_B, 1A6S, 1A7M, 1A91, 1A93_A, 1A93_B, 1A9V, 1AA3, 1AB3, 1AB7, 1ABT_A, 1ABV, 1ABZ, 1AC0, 1ACA, 1ACI, 1ADN, 1ADR, 1AF8, 1AFH, 1AFO_A, 1AGG, 1AH2, 1AH9, 1AHL, 1AIW, 1AJ3, 1AJE, 1AJW, 1AJY_A, 1AK6, 1AKP, 1AML, 1AO8, 1AOY, 1AP0, 1AP7, 1AP8, 1APC, 1APS, 1AQ5_A, 1ARB, 1AUU_A, 1AUZ, 1AW0, 1AW6, 1AWJ, 1AXH, 1AXJ, 1AYJ, 1AZ6, 1B16_A, 1B1A, 1B22_A, 1B4R_A, 1B64, 1B6F_A, 1B8O_A, 1B8W_A, 1B91_A, 1B9P_A, 1B9U_A, 1BA9, 1BAK, 1BAL, 1BAQ, 1BB8, 1BBG, 1BBI, 1BBN, 1BBY, 1BC6, 1BC9, 1BCI, 1BCT, 1BDC, 1BDS, 1BFM_A, 1BGF, 1BGK, 1BH4, 1BHU, 1BI6_H, 1BIP, 1BJ8, 1BJX, 1BKR_A, 1BKU, 1BL1, 1BLA, 1BLJ, 1BLR, 1BM4_A, 1BMR, 1BMW, 1BMX, 1BNO, 1BNR, 1BO0, 1BO9_A, 1BOE_A, 1BPR, 1BPV, 1BQV, 1BR0_A, 1BRV, 1BRZ, 1BSH_A, 1BT7, 1BUQ_A, 1BUY_A, 1BVE_A, 1BVH, 1BW3, 1BW6_A, 1BWX, 1BXD_A, 1BXO_A, 1BY1_A, 1BYI, 1BYM_A, 1BYQ_A, 1BYY_A, 1BZG, 1BZK_A, 1C01_A, 1C05_A, 1C0P_A, 1C1D_A, 1C1K_A, 1C20_A, 1C2N, 1C3Y_A, 1C4E_A, 1C55_A, 1C5E_A, 1C75_A, 1C7K_A, 1C7U_A, 1C89_A, 1C9Q_A, 1CCH, 1CCM, 1CDB, 1CDQ, 1CE4_A, 1CF4_B, 1CFE, 1CG7_A, 1CHC, 1CHL, 1CK2_A, 1CKV, 1CL4_A, 1CLH, 1CMO_A, 1CMR, 1CN2, 1CO4_A, 1COK_A, 1COO, 1COU_A, 1CUR, 1CW5_A, 1CWW_A, 1CWX_A, 1CX1_A, 1CYE, 1CYU, 1CZ4_A, 1D1D_A, 1D1H_A, 1D6G_A, 1D7Q_A, 1D8B_A, 1D8J_A, 1D8V_A, 1DAQ_A, 1DBD_A, 1DBF_A, 1DCI_A, 1DDF, 1DE1_A, 1DE3_A, 1DEC, 1DEF, 1DFE_A, 1DFS_A, 1DGF_A, 1DGN_A, 1DGQ_A, 1DIP_A, 1DJ0_A, 1DL0_A, 1DL6_A, 1DLX_A, 1DMC, 1DNY_A, 1DP3_A, 1DP7_P, 1DPU_A, 1DQB_A, 1DQC_A, 1DQZ_A, 1DRO, 1DS1_A, 1DS9_A, 1DTV_A, 1DU2_A, 1DU6_A, 1DUJ_A, 1DV0_A, 1DVH, 1DVJ_A, 1DWM_A, 1DX0_A, 1DX7_A, 1DX8_A, 1DXZ_A, 1DZ7_A, 1E01_A, 1E0A_B, 1E0E_A, 1E0H_A, 1E0L_A, 1E0Z_A, 1E17_A, 1E19_A, 1E29_A, 1E2B, 1E3T_A, 1E3Y_A, 1E4U_A, 1E53_A, 1E5G_A, 1E5U_I, 1E68_A, 1E6Q_M, 1E6U_A, 1E7L_A, 1E88_A, 1E8L_A, 1E8R_A, 1ECI_A, 1EDS_A, 1EDV_A, 1EDX_A, 1EF4_A, 1EGX_A, 1EH2, 1EHJ_A, 1EHS, 1EHX_A, 1EIK_A, 1EIT, 1EIW_A, 1EJ5_A, 1EKT_A, 1ELK_A, 1EMW_A, 1ENW_A, 1EO0_A, 1EO1_A, 1EP0_A, 1EQ3_A, 1EQO_A, 1ERC, 1ERD, 1ERX_A, 1ES9_A, 1ESX_A, 1EUW_A, 1EV0_A, 1EWI_A, 1EWS_A, 1EWW_A, 1EXE_A, 1EXG, 1EXK_A, 1EZA, 1EZG_A, 1EZO_A, 1EZT_A, 1F0Z_A, 1F24_A, 1F3C_A, 1F3R_B, 1F41_A, 1F53_A, 1F5Y_A, 1F81_A, 1F8P_A, 1FA3_A, 1FA4_A, 1FAF_A, 1FBR, 1FCT, 1FCY_A, 1FD8_A, 1FDM, 1FGP, 1FHO_A, 1FJ2_A, 1FJE_B, 1FJK_A, 1FJN_A, 1FM0_D, 1FMH_A, 1FO5_A, 1FP0_A, 1FQQ_A, 1FR3_A, 1FRE, 1FSH_A, 1FU9_A, 1FVL, 1FW9_A, 1FWO_A, 1FWP, 1FWQ_A, 1FYB_A, 1FYC, 1FYJ_A, 1FZT_A, 1G1E_B, 1G25_A, 1G26_A, 1G2H_A, 1G3G_A, 1G4F_A, 1G5V_A, 1G61_A, 1G66_A, 1G6E_A, 1G6S_A, 1G7D_A, 1G7E_A, 1G84_A, 1G90_A, 1G9L_A, 1GAB, 1GD0_A, 1GE9_A, 1GGW_A, 1GH9_A, 1GHC, 1GHH_A, 1GIO, 1GNC, 1GP8_A, 1GW3, 1GYF_A, 1H8C_A, 1HA9_A, 1HBW_A, 1HCD, 1HDO_A, 1HEV, 1HHN_A, 1HKS, 1HNR, 1HP8, 1HPW_A, 1HRE, 1HS7_A, 1HSQ, 1HX2_A, 1HYI_A, 1HYK_A, 1HYW_A, 1HZN_A, 1HZY_A, 1I0H_A, 1I1S_A, 1I25_A, 1I27_A, 1I5G_A, 1I5H_W, 1I5J_A, 1I6W_A, 1IBA, 1IBX_B, 1ICA, 1IHV_A, 1IIE_A, 1IJA_A, 1IL6, 1IMT, 1INZ_A, 1IOJ, 1IRF, 1IRL, 1IRP, 1IRS_A, 1ISU_A, 1ITF, 1IXH, 1JBA_A, 1JHB, 1JLI, 1JOY_A, 1JUN_A, 1JWE_A, 1KDX_A, 1KHM_A, 1KJS, 1KLA_A, 1KOE, 1KRS, 1KSR, 1LEA, 1LRE, 1LXL, 1LYP, 1MFN, 1MGS_A, 1MKC_A, 1MKN_A, 1MLA, 1MNT_A, 1MRO_B, 1MRO_C, 1MUN, 1MUT, 1MYF, 1NCS, 1NCT, 1NEQ, 1NGL_A, 1NGR, 1NKL, 1NLS, 1NOE, 1NS1_A, 1NTC_A, 1OAA, 1OLG_A, 1OM2_A, 1PA2_A, 1PAA, 1PCE, 1PCN, 1PCP, 1PEH, 1PFL, 1PFS_A, 1PIH, 1PIR, 1PLS, 1PMC, 1PMS, 1PNB_A, 1PNB_B, 1PNJ, 1PON_B, 1POU, 1PRR, 1PSM, 1QA5_A, 1QCE_A, 1QCK_A, 1QDP, 1QFD_A, 1QFQ_B, 1QFR_A, 1QFT_A, 1QGP_A, 1QH4_A, 1QHK_A, 1QJO_A, 1QK6_A, 1QK7_A, 1QK9_A, 1QKF_A, 1QKL_A, 1QKS_A, 1QL0_A, 1QLO_A, 1QM9_A, 1QN0_A, 1QND_A, 1QNR_A, 1QOP_B, 1QP6_A, 1QQF_A, 1QQI_A, 1QQV_A, 1QRJ_B, 1QRY_A, 1QSV_A, 1QTN_A, 1QTN_B, 1QTO_A, 1QTS_A, 1QTT_A, 1QTW_A, 1QU5_A, 1QU6_A, 1QYP, 1R2A_A, 1RAX_A, 1RCH, 1RCS_A, 1RES, 1RGE_A, 1RIE, 1RIP, 1ROT, 1RPR_A, 1RRB, 1RXR, 1SAP, 1SCY, 1SGG, 1SHC_A, 1SRO, 1SSN, 1SUH, 1SVF_A, 1SVF_B, 1SVQ, 1SWU_A, 1TBA_A, 1TBA_B, 1TBD, 1TBN, 1TFB, 1THF_D, 1TLE, 1TNS, 1TOF, 1TPM, 1TRL_A, 1TSG, 1U2F_A, 1UMS_A, 1URK, 1UTR_A, 1UWO_A, 1UXC, 1VGH, 1VPU, 1VRE_A, 1WDB, 1WFB_A, 1WHI, 1XBL, 1XNA_A, 1XNB, 1XPA, 1YGE, 1YUA, 1YUB, 1YUI_A, 1ZTA, 1ZTO, 2A3D_A, 2ALC_A, 2BID_A, 2CTC, 2END, 2EZH, 2EZK, 2EZM, 2FMR, 2GAT_A, 2GCC, 2GVA_A, 2HGF, 2HIR, 2HMX, 2IF1, 2IFE_A, 2IFO, 2JHB_A, 2LFB, 2LIS_A, 2MRB, 2NCM, 2NLR_A, 2OLB_A, 2ORC, 2PCF_B, 2PRF, 2PTH, 2PTL, 2REL, 2SOB, 2TMP, 2TPS_A, 2U2F_A, 2VIK, 3CHB_D, 3CRD, 3LRI_A, 3MEF_A, 3MSP_A, 3PHY, 3RPB_A, 3SIL, 3VUB, 3ZNF, 4EUG_A, 4ULL, 5GCN_A, 7A3H_A.

References

1. Davis, M. E.; McCammon, J. A. Chem Rev 1990, 90, 509–521.
2. Honig, B.; Nicholls, A. Science 1995, 268, 1144–1149.
3. Levy, R. M.; Gallicchio, E. Annu Rev Phys Chem 1998, 49, 531.
4. Cramer, C. J.; Truhlar, D. G. Chem Rev 1999, 99, 2161–2200.
5. Roux, B.; Simonson, T. Biophys Chem 1999, 78, 1–20.
6. Gilson, M. K. Curr Opin Struct Biol 1995, 5, 216–223.
7. Simonson, T. Curr Opin Struct Biol 2001, 11, 243–252.
8. Madura, J. D.; Davis, M. E.; Gilson, M. K.; Wade, R. C.; Luty, B. A.; McCammon, J. A. Rev Comput Chem 1994, 5.
9. Lee, B.; Richards, F. M. J Mol Biol 1971, 55, 379.
10. Sharp, K. A.; Honig, B. Annu Rev Biophys Biophys Chem 1990, 19, 301–332.
11. Harvey, S. C. Proteins 1989, 5, 78–92.
12. Schutz, C. N.; Warshel, A. Proteins 2001, 44, 400–417.

13. Davis, M. E.; McCammon, J. A. J Comput Chem 1991, 12, 909–912.
14. Im, W.; Beglov, D.; Roux, B. Comp Phys Comm 1998, 111, 59–75.
15. Grant, J. A.; Pickup, B. T.; Nicholls, A. J Comput Chem 2001, 22, 608–640.
16. Fogolari, F.; Brigo, A.; Molinari, H. J Mol Recog 2002, 15, 377–392.
17. Warwicker, J.; Watson, H. C. J Mol Biol 1982, 157, 671–679.
18. Madura, J. D.; Briggs, J. M.; Wade, R.; Davis, M. E.; Luty, B. A.; Ilin, A.; Antonsiewicz, J.; Gilson, M. K.; Bagheri, B.; Scott, L. R.; McCammon, J. A. Comp Phys Comm 1995, 91, 57–95.
19. Gilson, M. K.; Sharp, K. A.; Honig, B. H. J Comput Chem 1987, 9, 327–335.
20. Rocchia, W.; Sridharan, S.; Nicholls, A.; Alexov, E.; Chiabrera, A.; Honig, B. J Comput Chem 2002, 23, 128–137.
21. Baker, N.; Holst, M.; Wang, F. J Comput Chem 2000, 21, 1343–1352.
22. Holst, M.; Baker, N.; Wang, F. J Comput Chem 2000, 21, 1319–1342.
23. Cortis, C. M.; Friesner, R. A. J Comput Chem 1997, 18, 1570–1590.
24. Cortis, C. M.; Friesner, R. A. J Comput Chem 1997, 18, 1591–1608.
25. Rashin, A. A.; Namboodiri, K. J Phys Chem 1987, 91, 6003–6012.
26. Miertus, S.; Scrocco, E.; Tomasi, J. Chem Phys 1981, 55, 117–129.
27. Zauhar, R. J.; Morgan, R. S. J Mol Biol 1985, 186, 815–820.
28. Boschitsch, A. H.; Fenley, M. O.; Zhou, H.- X. J Phys Chem B 2002, 106, 2741–2754.
29. Bordner, A. J.; Huber, G. A. J Comput Chem 2003, 24, 353–367.
30. Bashford, D.; Karplus, M. Biochemistry 1990, 29, 10219–10225.
31. Yang, A. S.; Gunner, M. R.; Sampogna, R.; Sharp, K.; Honig, B. Proteins 1993, 15, 252–265.
32. Antosiewicz, J.; McCammon, J. A.; Gilson, M. K. J Mol Biol 1994, 238, 415–436.
33. Dominy, B. N.; Perl, D.; Schmid, F. X.; Brooks, C. L. III. J Mol Biol 2002, 319, 541–554.
34. Srinivasan, J.; Cheatham, T. E.; Cieplak, P.; Kollman, P. A.; Case, D. A. J Am Chem Soc 1998, 120, 9401–9409.
35. Lee, M. R.; Duan, Y.; Kollman, P. A. Proteins 2000, 39, 309–316.
36. Lee, M. R.; Kollman, P. A. Structure 2001, 9, 905–916.
37. Zauhar, R. J. J Comput Chem 1991, 12, 575–583.
38. Sharp, K. J Comput Chem 1991, 12, 454–468.
39. Niedermeier, C.; Schulten, K. Mol Simul 1992, 8, 361–387.
40. Gilson, M. K. J Comput Chem 1995, 16, 1081–1095.
41. Luo, R.; David, L.; Gilson, M. K. J Comput Chem 2002, 23, 1244–1253.
42. Lu, B. Z.; Chen, W. Z.; Wang, C. X.; Xu, X.- J. Proteins 2002, 48, 497–504.
43. Wan, S. Z.; Wang, C. X.; Xiang, Z. X.; Shi, Y. Y. J Comput Chem 1997, 18, 1440–1449.
44. Friedrichs, M.; Zhou, R. H.; Edinger, S. R.; Friesner, R. A. J Phys Chem B 1999, 103, 3057–3061.
45. Bashford, D.; Case, D. A. Annu Rev Phys Chem 2000, 51, 129–152.
46. Born, M. Z Phys 1920, 1, 45–48.
47. Constanciel, R.; Contreras, R. Theor Chim Acta 1984, 65, 1–11.
48. Still, W. C.; Tempczyk, A.; Hawley, R. C.; Hendrickson, T. J Am Chem Soc 1990, 112, 6127–6129.
49. Srinivasan, J.; Trevathan, M. W.; Beroza, P.; Case, D. A. Theor Chim Acta 1999, 101, 426–434.
50. Onufriev, A.; Case, D. A.; Bashford, D. J Comput Chem 2002, 23, 1297–1304.
51. Qiu, D.; Shenkin, P. S.; Hollinger, F. P.; Still, W. C. J Phys Chem A 1997, 101, 3005–3014.
52. Tsui, V.; Case, D. A. Biopolym Nucl Acid Sci 2001, 56, 275–291.
53. Onufriev, A.; Bashford, D.; Case, D. A. J Phys Chem B 2000, 104, 3712–3720.
54. Jayaram, B.; Liu, Y.; Beveridge, D. L. J Chem Phys 1998, 109, 1465–1471.
55. Dominy, B. N.; Brooks, C. L. III. J Phys Chem B 1999, 103, 3765–3773.
56. Hawkins, G. D.; Cramer, C. J.; Truhlar, D. G. Chem Phys Lett 1995, 246, 122–129.
57. Schaefer, M.; Karplus, M. J Phys Chem 1996, 100, 1578–1599.
58. Schaefer, M.; Bartels, C.; Leclerc, F.; Karplus, M. J Comput Chem 2001, 22, 1857–1879.
59. Ghosh, A.; Rapp, C. S.; Friesner, R. A. J Phys Chem B 1998, 102, 10983–10990.
60. Lee, M. S.; Feig, M.; Salsbury, F. R. Jr.; Brooks, C. L. III. J Comput Chem 2003, 24, 1348–1356.
61. Lee, M. S.; Salsbury, F. R. Jr.; Brooks, C. L. III. J Chem Phys 2002, 116, 10606–10614.
62. Jayaram, B.; Sprous, D.; Beveridge, D. L. J Phys Chem B 1998, 102, 9571–9576.
63. Dominy, B. N.; Brooks, C. L. III. J Comput Chem 2001, 23, 147–160.
64. Feig, M.; Brooks, C. L. III. Proteins 2002, 49, 232–245.
65. Felts, A. K.; Gallicchio, E.; Wallqvist, A.; Levy, R. M. Proteins 2002, 48, 404–422.
66. Zhang, L. Y.; Gallicchio, E.; Friesner, R. A.; Levy, R. M. J Comput Chem 2001, 22, 591–607.
67. Tsui, V.; Case, D. A. J Am Chem Soc 2000, 122, 2489–2498.
68. Calimet, N.; Schaefer, M.; Simonson, T. Proteins 2001, 2001, 144–158.
69. Zhu, J.; Shi, Y.; Liu, H. J Phys Chem B 2002, 106, 4844–4853.
70. Krol, M. J Comput Chem 2003, 24, 531–546.
71. Wang, T.; Wade, R. C. Proteins 2003, 50, 158–169.
72. Nina, M.; Simonson, T. J Phys Chem B 2002, 106, 3696–3705.
73. David, L.; Luo, R.; Gilson, M. K. J Comput Chem 2000, 21, 295–309.
74. MacKerell, A. D. Jr.; Bashford, D.; Bellott, M.; Dunbrack, J. D.; Evanseck, M. J.; Field, M. J.; Fischer, S.; Gao, J.; Guo, H.; Ha, S.; Joseph-McCarthy, D.; Kuchnir, L.; Kuczera, K.; Lau, F. T. K.; Mattos, C.; Michnick, S.; Ngo, T.; Nguyen, D. T.; Prodhom, B.; Reiher, W. E.; Roux, B.; Schlenkrich, M.; Smith, J. C.; Stote, R.; Straub, J.; Watanabe, M.; Wiorkiewicz-Kuczera, J.; Yin, D.; Karplus, M. J Phys Chem B 1998, 102, 3586–3616.
75. Cornell, W. D.; Cieplak, P.; Bayly, C. I.; Gould, I. R.; Merz, K. M.; Ferguson, D. M.; Spellmeyer, D. C.; Fox, T.; Caldwell, J. W.; Kollman, P. A. J Am Chem Soc 1995, 117, 5179–5197.
76. Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. J Am Chem Soc 1996, 118, 11225–11236.
77. Brooks, B. R.; Bruccoleri, R. E.; Olafson, B. D.; States, D. J.; Swaminathan, S.; Karplus, M. J Comput Chem 1983, 4, 187–217.
78. Berman, H. M.; Westbrook, J.; Feng, Z.; Gilliland, G.; Bhat, T. N.; Weissig, H.; Shindyal, I. N.; Bourne, P. E. Nucl Acids Res 2000, 28, 235–242.
79. Kolinski, A.; Skolnick, J. Proteins 1998, 32, 475–494.
80. Feig, M.; Rotkiewicz, P.; Kolinski, A.; Skolnick, J.; Brooks, C. L. III. Proteins 2000, 41, 86–97.
81. Onufriev, A.; Case, D. A.; Bashford, D. J Mol Biol 2003, 325, 555–567.
82. Roux, B. Biophys J 1997, 73, 2980–2989.
83. Nina, M.; Beglov, D.; Roux, B. J Phys Chem 1997, 101, 5239–5248.
84. Nicholls, A.; Honig, B. J Comput Chem 1991, 12, 435–445.
85. Abramowitz, M.; Stegun, I. A. Handbook of Mathematical Functions, With Formulas, Graphs, and Mathematical Tables; Dover Publications: New York, 1974.
86. Lebedev, V. I.; Laikov, D. N. Doklady Math 1999, 59, 477–481.

87. Im, W.; Lee, M. S.; Brooks, C. L. III. J Comput Chem 2003, 24, 1691–1700.
88. Press, W. H.; Flannery, B. P.; Teukolsky, S. A.; Vetterling, W. T. Numerical Recipes in C; Cambridge University Press: Cambridge, UK, 1988.
89. Hawkins, G. D.; Cramer, C. J.; Truhlar, D. G. J Phys Chem 1996, 100, 19824–19839.
90. Bondi, A. J Phys Chem 1964, 68, 441–451.
91. Ponder, J. W.; Richards, F. M. J Comput Chem 1987, 8, 1016–1024.
92. Kitchen, D. B.; Hirata, F.; Westbrook, J. D.; Levy, R.; Kofke, D.; Yarmush, M. J Comput Chem 1990, 10, 1169–1180.
93. Bashford, D., ed. An Object-Oriented Programming Suite for Electrostatic Effects in Biological Molecules, vol. 1343; Springer: Berlin, 1997.
94. Gilson, M. K.; Davis, M. E.; Luty, B. A.; McCammon, J. A. J Phys Chem 1993, 97, 3591–3600.
95. Totrov, M.; Abagyan, R. Biopolyms Peptide Sci 2001, 60, 124–133.
96. Abagyan, R. A.; Totrov, M. M.; Kuznetsov, D. A. J Comput Chem 1994, 15, 488–506.
97. Sitkoff, D.; Sharp, K. A.; Honig, B. J Phys Chem 1994, 98, 1978–1988.
98. Jang, S.; Shin, S.; Pak, Y. J Am Chem Soc 2002, 124, 4976–4977.
99. Pak, Y.; Jang, S.; Shin, S. J Chem Phys 2002, 116, 6831–6835.
100. Masunov, A.; Lazaridis, T. J Am Chem Soc 2003, 125, 1722–1730.