JCTC Journal of Chemical Theory and Computation

Article

pubs.acs.org/JCTC

PowerBorn: A Barnes–Hut Tree Implementation for Accurate and Efficient Born Radii Computation

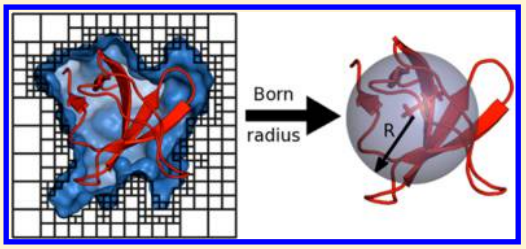
Martin Brieg[†] and Wolfgang Wenzel*[,‡]

[†]Steinbuch Centre for Computing (SCC), Karlsruhe Institute of Technology (KIT), P.O. Box 3640, 76021 Karlsruhe, Germany

[‡]Institute of Nanotechnology (INT), Karlsruhe Institute of Technology (KIT), P.O. Box 3640, 76021 Karlsruhe, Germany

Supporting Information

ABSTRACT: Implicit solvent models are one of the standard tools in computational biophysics. While Poisson–Boltzmann methods offer highly accurate results within this framework, generalized Born models have been used due to their higher computational efficiency in many (bio)molecular simulations, where computational power is a limiting factor. In recent years, there have been remarkable advances to reduce some deficiencies in the generalized Born models. On the other hand, these advances come at an increased computational cost that contrasts the reasons for choosing generalized Born models over Poisson–Boltzmann methods. To address this performance issue, we present a new algorithm for Born radii computation, one performance critical part in the evaluation of generalized Born models, which is based on a Barnes–Hut tree code scheme. We show that an implementation of this algorithm provides accurate Born radii and polar solvation free energies in comparison to Poisson–Boltzmann computations, while delivering up to an order of magnitude better performance over existing, similarly accurate methods. The C++ implementation of this algorithm will be available at http://www.int.kit.edu/nanosim/.



INTRODUCTION

While solvation effects play a central role in many biomolecular[1] and chemical[2] processes, the details of the solvent behavior itself are typically of minor interest compared to the effect of the solvent on the solute.[3] Even though an explicit solvent representation offers a very accurate description of the system, it is often impractical to use such a representation in computer simulations due to the computational cost that is associated with the high number of degrees of freedom in such a system.[4] Implicit solvent models integrate out the solvent degrees of freedom, resulting in a potential of mean force that describes the effect of the solvent on the solute.[3] Over the years, many different models have been developed,[5−8] which vary widely in the approaches and methods employed, the resulting accuracy, and the targeted applications.

In molecular biophysics, models based on the Poisson–Boltzmann (PB) theory[9−12] are an established standard tool with many applications.[9,13−22] Due to the considerable cost of solving the PB equation numerically and the size of biologically interesting systems, one may also resort to a large number of computationally less expensive approximations based on the generalized Born (GB) model,[9,23−34] which have become common in applications like dynamics studies of proteins[35,36] and nucleic acids,[37,38] protein–protein docking,[39] or small peptide folding simulations[40] among many others.

Evaluation of the polar part of the solvation free energy in the GB model requires computation of so-called atomic Born radii, which describe the amount of polarization that is induced by the partial charge of an atom in the dielectric environment. Using the Coulomb-field approximation, Born radii R_i may be computed by solving the following integral

$$\frac{1}{R_i} = \frac{1}{4\pi}\int_{\text{water}} \frac{dV}{\| \vec{r}_i - \vec{r} \|^4} \tag{1}$$

where the integration region is the volume outside the molecule occupied by water and $\vec{r}_i$ is the position of the atom in question. Given these Born radii, the polar part of the solvation free energy can then be computed using Still's formula[32]

$$\Delta G = -\frac{\alpha}{2}\left(\frac{1}{\varepsilon_p} - \frac{1}{\varepsilon_w}\right)\sum_{i,j}\frac{q_i q_j}{\sqrt{r_{ij}^2 + R_i R_j \exp(-r_{ij}^2/4R_iR_j)}} \tag{2}$$

where α = 331.84 kcal/mol is a constant, ε_p and ε_w are the dielectric constants of the protein and water, respectively, q denotes the partial charge of an atom, and r_{ij} is the distance between two atoms. While implementation of eq 2 is straightforward and can be combined with evaluation of the Coulomb interaction, computing the integral in eq 1 is more challenging due to the complex integration region that is defined by the molecular conformation and the chosen surface definition.

Since PB calculations with the solvent excluded surface (SES) definition[41−43] yield good results in comparison to hybrid[44] or explicit[45] solvent simulations, efforts were made to incorporate the effects of the more complex SES by using

Received: October 8, 2012
Published: February 4, 2013

modifications to the Born radii typically computed from the simpler van der Waals surface[46] or adding volume corrections to the integral of the van der Waals surface to account for small water filled cavities.[47,48] Other methods use atom-centered functions to define the surface and apply a numerical volume integration scheme.[49−51] Models where the integration is carried out directly over the surface itself instead of the volume have also been proposed.[29,34,52]

Another focus of improvements has been the integral expression by which the Born radii are computed. Higher order corrections to eq 1 have been proposed[50,53] to compensate shortcomings of the Coulomb-field approximation, on which the integral is based. Using Kirkwood's[54] work, Grycuk[55] also derived a new integral expression

$$\frac{1}{R_i^3} = \frac{3}{4\pi} \int_{\text{water}} \frac{dV}{\| \vec{r}_i - \vec{r} \|^6} \tag{3}$$

which was shown to yield accurate Born radii if the same surface as in PB calculations is used.[56]

While the progress in accuracy has been remarkable so far, the corrections and additional contributions lead to an increased computational cost. Therefore, simulations with models like GBMV2,[49] which is considered one of the most accurate in comparison to PB calculations with the SES,[57] are about 12 times more expensive than other GB models.[58] Since the GB term dominates the cost of biomolecular simulations using an implicit-solvent representation, it is important to develop a method that combines computation of very accurate Born radii with a numerically highly efficient implementation. However this challenge has not yet been overcome.[59]

To address this issue we present a new method for Born radii computation called PowerBorn along with an implementation focusing on performance as well as accuracy. Our approach aims primarily at applications like large-scale simulations of biomolecular systems using Monte Carlo methods, which are also suitable to study protein folding or large scale conformational changes in biomolecules[60] but may be extended to other applications as well. On the basis of a Barnes−Hut tree code scheme,[61] we developed a novel approach to perform the numerical integration of the integral expression for Born radii and demonstrate the accuracy of our method by comparing it to results from PB computations. The performance of our implementation is assessed by speedup measurements with two other state-of-the-art methods used for Born radii computation.

We first begin by describing our algorithm, regarding especially the construction of the octree representing the molecule and its surrounding water, and the determination of the approximate SES. After explaining details of the integration scheme, the extensions needed to also implement gradients for molecular dynamics simulation are outlined. Finally, the accuracy and performance of our implementation are assessed and discussed.

METHODS

To compute the Born radii, we will rely on Grycuk's integral in eq 3, but in contrast to most other volume integration methods, we will perform the integration over the water region and not convert the integral to that over the molecular region. Since most proteins form globular shapes, the integration region is typically far away from the atoms, except for those at the surface, thus making it easier to apply approximations to the rapidly decreasing integrand.

To take advantage of this, our algorithm will be split into two steps. The first step consists of constructing an octree representation of the water region around the molecules' surface inside a bounding box. The second step performs the integration by a walk through the octree similar to the Barnes−Hut tree scheme,[61] adding the contribution of a cell to the integral whenever a given accuracy criterion is met. For buried atoms, the walk will end rather high up in the octree, thus providing fast integration, while for atoms at the surface, the walk will go deeper into the octree to reach the needed accuracy. Finally, the region outside the bounding box is integrated via analytical formulas.

We note that both steps are independent from each other and only the octree construction depends on the surface of the molecule, making it easy to implement various surface definitions into this framework. In the next sections, the octree construction and integration will be explained in more detail.

Octree Construction and Molecular Surface Approximation. An octree is a tree structure used to subdivide a region of three-dimensional space and store data related to these subregions. The nodes of this tree structure are represented by cubic cells, where the bounding box of our problem will be referred to as the root cell. We will characterize these cells by their center $\vec{r}_c$ and their edge length s_c. Each tree node may have eight branches, hence the name octree, which are the eight equal sized cubic cells that take up the same volume and will be referred to as the children of that cell. For each cell, we will store the volume outside the molecules' surface, e.g., the volume V occupied by water in that cell, and the corresponding centroid $\vec{c}$ of that volume, which will be used later to solve the integrals for the Born radii.

For this work, we have chosen to implement the SES definition for the reasons given in the Introduction. We approximate it by sampling a finite number of points on the solvent accessible surface (SAS)[62] and place water spheres with the size of the probe radius p_w on these points. The points are sampled so that the nearest neighbor distance d fulfills the following criterion:

$$d < wp_w \tag{4}$$

where w is an input parameter determining the amount of sampling points and therefore the accuracy of the approximation to the SES. Everything inside the SAS and outside these water spheres will then be considered enclosed by our SES approximation.

Recently, a method for computing the SAS area and volume based on a power diagram representation of the conformation was published, yielding accurate and fast results.[63] Because these terms are typically used to model nonpolar solvation effects, we will also use the precomputed power diagram to enhance speed and accuracy of our Born radii method. From the power diagram, we derive which atoms are completely covered by others, thus removing the need to create and check sampling points for these atoms. We also use the so-called surface vertices of the power diagram, points at which three spheres of the SAS intersect, as sampling points. Since these are typically located in crevices at the SAS, they cover a lot of volume enclosed by the SAS, which would otherwise falsely be considered to belong to the molecule, leading to an overestimation of Born radii.

The octree construction starts with finding a bounding box of the molecule, so that the edge size of the smallest cell in the octree will equal the resolution specified by the user in the

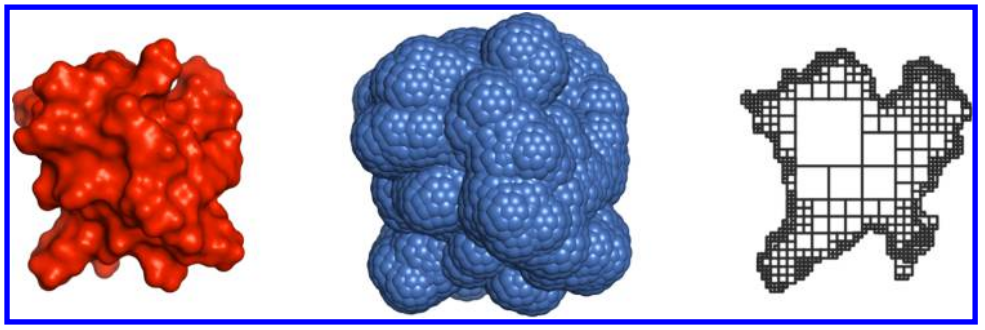


Figure 1. Solvent excluded surface (SES) of PDB 1SHG[64] (left), water spheres used by PowerBorn to approximate the SES (center), and a slice of the octree representing the solute volume (right).

program input. Starting with the bounding box, we determine for each cell the atoms overlapping with it. If no water spheres overlap with a cell, that cell is considered completely inside or outside the molecule, depending on whether any spheres of which the SAS consist reach into that cell, and therefore the construction of the present branch of the octree is completed. If only water spheres reach into the cell, or it is completely buried within one single water sphere, the cell is considered completely outside the molecule. If none of the former conditions are met, eight child cells are constructed, and the procedure is recursively repeated for those until a decision is made, or the smallest cell size is reached. For this finest level of resolution, a cell is considered completely inside the molecule, if its center is inside the SAS and not inside any water sphere; otherwise it is considered completely outside the molecule.

After the octree construction is completed, we discard all children of a cell, if all of them are either inside or outside our surface. The resulting octree representing the solute volume is visualized in Figure 1, together with the SES and the water spheres for the SH3 domain of *Escherichia coli*[64] (PDB code: 1SHG). In the last step, each octree cell is recursively assigned the volume V in the cell being outside the molecule and the corresponding centroid $\vec{c}$ of that volume

$$V = \sum_{\text{children } k} V_k,\ \vec{c} = \frac{1}{V} \sum_{\text{children } k} \vec{c}_k V_k \tag{5}$$

Octree Integration. For each atom, a top-down walk through the octree is performed. In analogy to Barnes and Hut,[61] a cell's contribution to a Born radius of an atom at position $\vec{r}_i$ will be added to the integral if the centroid $\vec{c}$ and size s_c of that cell meet the following criterion

$$\| \vec{r}_i - \vec{c} \|^2 > \frac{s_c^2 f}{4} \tag{6}$$

where the integration factor f is an input parameter determining the speed and accuracy of the integration. If the criterion holds, the contribution ΔI to the integral in eq 3 is computed by the zeroth-order Taylor expansion of the integrand using the volume V of water in the cell from eq 5

$$\Delta I = \frac{V}{\| \vec{r}_i - \vec{c} \|^6} \tag{7}$$

In case the criterion is not met, the algorithm recursively proceeds to all children having volume outside the molecule until the criterion is met, or the cell has no further children. In the latter case, a numerical plain grid integration is performed over that cell instead of using eq 7.

The integration over the region outside the bounding box is done by converting the volume integral via Gauss' law to integrals over the faces of the bounding box

$$I_{\text{Box}} = \int_V \frac{dV}{\| \vec{r}_i - \vec{r} \|^6} = \oint_{\partial V} \frac{(\vec{r}_i - \vec{r})}{3 \| \vec{r}_i - \vec{r} \|^6} \cdot d\vec{S} \tag{8}$$

These integrals can be solved analytically, e.g., for the top square of a cube with its center at $\vec{r}_c$, edge length s_c, normal vector pointing in the negative z direction since we want to integrate over the region outside the cube, and an atom at $\vec{r}_i$

$$I_{\text{Top}} = \frac{1}{3} \int_{x_l}^{x_u} \int_{y_l}^{y_u} \frac{z_u}{(x^2 + y^2 + z_u^2)^3}\, dx\, dy \tag{9}$$

where we have inserted the upper and lower integration boundaries

$$\begin{aligned} x_{\text{u,l}} &= \pm\frac{s_c}{2} - x_i + x_c, \\ y_{\text{u,l}} &= \pm\frac{s_c}{2} - y_i + y_c, \\ z_{\text{u,l}} &= \pm\frac{s_c}{2} - z_i + z_c \end{aligned} \tag{10}$$

The solution using the primitive function F is given by

$$\begin{aligned} I_{\text{Top}} = \frac{z_u}{3} \Bigg(& F\left(z_u, \frac{y_u}{z_u}, \frac{x_u}{z_u}\right) - F\left(z_u, \frac{y_l}{z_u}, \frac{x_u}{z_u}\right) \\ & - F\left(z_u, \frac{y_u}{z_u}, \frac{x_l}{z_u}\right) + F\left(z_u, \frac{y_l}{z_u}, \frac{x_l}{z_u}\right) \Bigg) \end{aligned} \tag{11}$$

$$\begin{aligned} F(x, y, z) = \frac{1}{8x^4} \Bigg(& \frac{yz(2 + y^2 + z^2)}{(1 + y^2)(1 + z^2)(1 + y^2 + z^2)} \\ & + g(y, z) + g(z, y) \Bigg) \end{aligned} \tag{12}$$

$$g(y, z) = \frac{y(3 + 2y^2) \arctan(z/\sqrt{1 + y^2})}{(1 + y^2)^{3/2}} \tag{13}$$

As expected, the primitive function is symmetric in y and z, which can be exploited to optimize performance when computing such a complex formula. The expression for the integral of the whole cube is given in the Supporting Information. Applying eq 3 to the computed integrals then yields the Born radii.

Extensions for Gradients. As described earlier, our implementation is aimed for simulations using Monte Carlo

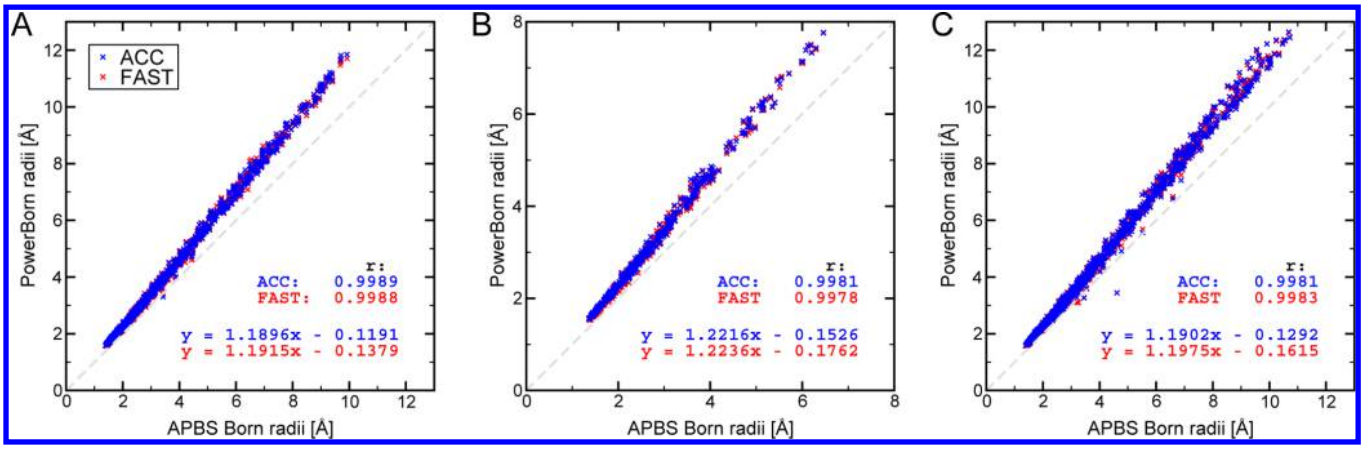


Figure 2. Comparison of Born radii from APBS[66] reference calculations to PowerBorn radii for the two input parameter sets ACC and FAST for PDBs 1SHG[64] (A), 1AZ6[67] (B), and 1UBQ[68] (C). Pearson correlation coefficients and least-squares fits are also shown in the plots.

methods, for which gradients are not required. Because of the great popularity of molecular dynamics in molecular biophysics, we will describe here the parts necessary to implement gradients to our octree based algorithm.

As with any other grid based method in molecular mechanics, one needs to take special care of discontinuities and discretization errors, but also model and simulation parameters, to achieve stable trajectories.[65] Consequently, one should choose a different surface definition in order to avoid problems arising from discontinuous changes in the SES over time,[28] e.g., when cavities are temporarily formed. Also, the decision if a smallest octree cell is inside or outside, which for our algorithm is discontinuous, should be replaced, so that the fraction of the volume covered by water and the corresponding centroid change smoothly as a function of the position of the cell in relation to the molecular surface. How exactly this smooth criterion looks depends on the choice of the smooth surface. As stated in the original paper by Barnes and Hut,[61] their force error, and accordingly the error of the Born radii and gradients in this work, will depend on the input parameters such as the octree resolution and the integration factor of eq 6, which therefore should be carefully chosen to limit the numerical force errors as one deems necessary.

To compute forces from eq 2, the derivative of the Born radius of an atom i with respect to the coordinates of some atom j are needed. Using algebraic manipulations, one finds that the derivatives of the integral for the Born radii in respect to the coordinates of a given atom are required. Approximating the integral in eq 3 to a finite sum over the octree cells passing the accuracy criterion, and applying the derivative, we end up with

$$\nabla_j \frac{3}{4\pi}\int_{\text{water}} \frac{dV}{\|\vec{r}_i - \vec{r}\|^6} \approx \frac{3}{4\pi}\sum_{\substack{k=\text{accepted}\\ \text{cells}}} V_k\left(\frac{-3}{\|\vec{r}_i - \vec{c}_k\|^8}\nabla_j(\vec{r}_i - \vec{c}_k)^2\right) + \frac{1}{\|\vec{r}_i - \vec{c}_k\|^6}(\nabla_j V_k) \quad (14)$$

In addition to the volume in a cell covered by water and the corresponding centroid, one also needs to compute their gradients with respect to the coordinates of the atoms in the molecule. The details of this rely heavily on the chosen surface definition; however for a cell that has no children, these gradients are only nonvanishing if the cell is near the surface of the molecule and the respective atoms are in close proximity to that cell. The gradients of the volume and centroid also need to be computed for the parent cells. According to eq 5, the volume gradient for the parent cell is simply given by the sum of the children volume gradients, while the expression for the centroid is more complex. This could be avoided by just using the center of a cell, which does not depend on the position of the atoms, instead of the centroid, so that only the volume gradients are needed. However, this simplified algorithm may require a finer octree resolution and a higher integration factor to limit discretization errors, therefore degrading performance of the algorithm.

RESULTS AND DISCUSSION

Comparison to Poisson–Boltzmann Results. To test the accuracy of our algorithm, we have employed two parameter sets. The first set corresponds to a more accurate version (further referred to as ACC), with the smallest octree cell having an edge length of 0.25 Å, the same as the grid size in the reference PB calculations, and an integration factor of f = 10.0. This should demonstrate the accuracy limit of the GB model. The resolution of the octree necessary to reach good agreement with PB results will also depend on the details of the surface, e.g., the chosen set of atomic radii, where smaller radii require a better octree resolution. For details on the radii set used in this work, please refer to the Appendix. The second set is intended for applications with an octree resolution of 0.4 Å on the lowest level and an integration factor of f = 8.0 (further referred to as FAST).

To assess the accuracy of our suggested algorithm, we first compare the Born radii computed with the PB solver APBS[66] to our PowerBorn radii. Since computing Born radii from PB results requires two accurate solutions of the PB equation for each Born radius, one for the vacuum state and one for the solvated state, this test is computationally intensive. Therefore, we have restricted ourselves to performing this analysis only for three structures, for which we have arbitrarily chosen the SH3 domain of *Escherichia coli*[64] (PDB code: 1SHG), an engineered cellulose-binding domain of cellobiohydrolase I from *Trichoderma reesei*[67] (PDB code: 1AZ6), and ubiquitin[68] (PDB code:

1UBQ). Figure 2 shows the scatter plots for the computed Born radii from APBS and our PowerBorn algorithm.

We observe a very good correlation between the two data sets with Pearson correlation coefficients of at least 0.9981 for ACC and 0.9978 for FAST. We note that for these structures, the FAST version is marginally less accurate than ACC, except for PDB 1UBQ, where FAST has a slightly higher correlation coefficient than ACC, proving that the parameters chosen for FAST still yield accurate Born radii when compared to PB computations.

Contrary to the very good results for the correlation coefficients, the linear fits show a systematic overestimation of our PowerBorn radii that is proportional to the size of the Born radius. We have investigated this discrepancy closer and find the cause for this behavior in eq 7, which systematically underestimates the integral of a completely water filled cube when compared to the analytic solution using eqs 8−13. This is also true for partly water filled cubes, but with a smaller systematic deviation. This issue can be alleviated by using a modified version of eq 7

$$\Delta I = \frac{\beta V}{(\| \vec{r}_i - \vec{c} \|^2 + \alpha s_c^2)^3} \tag{15}$$

where α and β are fit parameters and s_c is the edge length of the cube. However, at this stage, the formula offers no clear advantage, since it only changes the linear fit but does not improve the correlation significantly to justify the additional computational cost (data not shown).

For 1UBQ, there are also a few outliers in the comparison of APBS and PowerBorn radii, where PowerBorn seems to underestimate the Born radius. The three atoms with the largest deviation from the linear regression given in Figure 2C are the HB3 atom of residue arginine 42, HG12 of valine 5, and the O atom of valine 70. For the first and third case, we find that both atoms are close to a spot where the PowerBorn surface differs from the APBS surface, with the former being much closer to the two atoms (see Supporting Information for a visualization of the surface). This tighter surface is the reason for the smaller Born radii in these two cases. In the second case the surface deviation is smaller, but the atom is located near a deep invagination, which is the only dominant source of solvation for this otherwise buried atom (again see Supporting Information for a visualization). Therefore, the Born radius of that atom is very sensitive to even small changes in the surface of that invagination. We also note that in contrast to the results of Mongan et al.[56] we do not observe this invagination to cause an overestimation of the Born radius because of the different surfaces of the GB and PB methods used in this work.

Since Born radii are only an intermediate step to compute accurate polar solvation free energies via eq 2, we have also investigated other fit methods than those shown in Figure 2 to improve accuracy of the computed energies instead of the Born radii. Due to the inverse Born radii being proportional to the self-polarization free energy

$$\Delta G_i^{\text{self}} = -\frac{\alpha}{2}\left(\frac{1}{\varepsilon_p} - \frac{1}{\varepsilon_w}\right)\frac{q_i^2}{R_i} \tag{16}$$

which are the self-energy terms of eq 2, our fit will have the following form

$$\frac{1}{\tilde{R}_i} = a\frac{1}{R_i} + b \tag{17}$$

where the parameters a and b will be optimized to minimize the relative root-mean-square error of the polar solvation free energies for the protein structures in the training set compared to APBS reference computations. For the training and test sets, we have chosen those published by Feig et al.,[69] which consist of 22 and 611 protein structures, respectively. Using Python scipy's Powell minimization algorithm, we find the values shown in Table 1 for the two parameters.

Table 1. Values Obtained from the Training Set for the Two Fit Parameters in eq 17

parameter set	a	b
ACC	1.0667862	0.03516313
FAST	1.04697536	0.03994111

Before we have a closer look at the test set, we examine the effects of this fit to the Born radii of PDB 1SHG a little bit closer. Figure 3 shows the corresponding Born radii from Figure 2 as well as their inverse and the computed self-polarization energies after applying the linear fit in eq 17. Interestingly, the fit weakens the correlation between APBS and our Born radii, because the nonvanishing parameter b causes the PowerBorn radii to saturate as a function of b^{-1}. We suspect that the fit to the polar solvation free energies leads to this behavior to cancel a shortcoming of eq 2 which may induce a systematical error for large Born radii. The deviation is also apparent for the inverse Born radii, where small values are always larger than those computed with APBS. In contrast, this behavior is not as visible in the self-polarization energy plot, where the partial charges cause the systematic overestimation of small inverse Born radii to be scattered over a large range of self-polarization energies, thus effectively hiding the systematic error. We note that with a Pearson correlation coefficient of 0.9994, our FAST method is still as accurate as GBMV2, for which a similar high correlation coefficient was reported for a test set of small protein structures.[49]

We have next computed polar solvation free energies for all structures in the test set using APBS and PowerBorn radii together with eq 2. A histogram of the relative errors is shown in Figure 4, and an analysis is given in Table 2. The relative average unsigned error is 0.66% and 0.68% for ACC and FAST, respectively, again demonstrating that FAST yields nearly as good results as ACC. Although the relative root-mean-square errors are still small with about 1%, they are nearly half again as large as the average unsigned errors, because they are more sensitive to few large outliers. The largest of these outliers for both parameter sets is concanavilin A[70] (PDB code: 1NLS), with relative errors of 8.18% and 8.16% for ACC and FAST respectively.

We have examined this structure closer and find that, with the given solvent probe radius and atomic radii to define the SES, this structure possesses several cavities and a tunnel, as shown in Figure 5. Mongan et al. showed that for these cases the integral expression used to determine the Born radii does suffer from systematic errors;[56] thus the large error in comparison to the PB result is due to limitations of the underlying GB model and not due to our algorithm. This is also consistent with the fact that both input parameter sets show this large error for the same structure, but do agree very well with each other on the absolute polar solvation free energy of this structure.

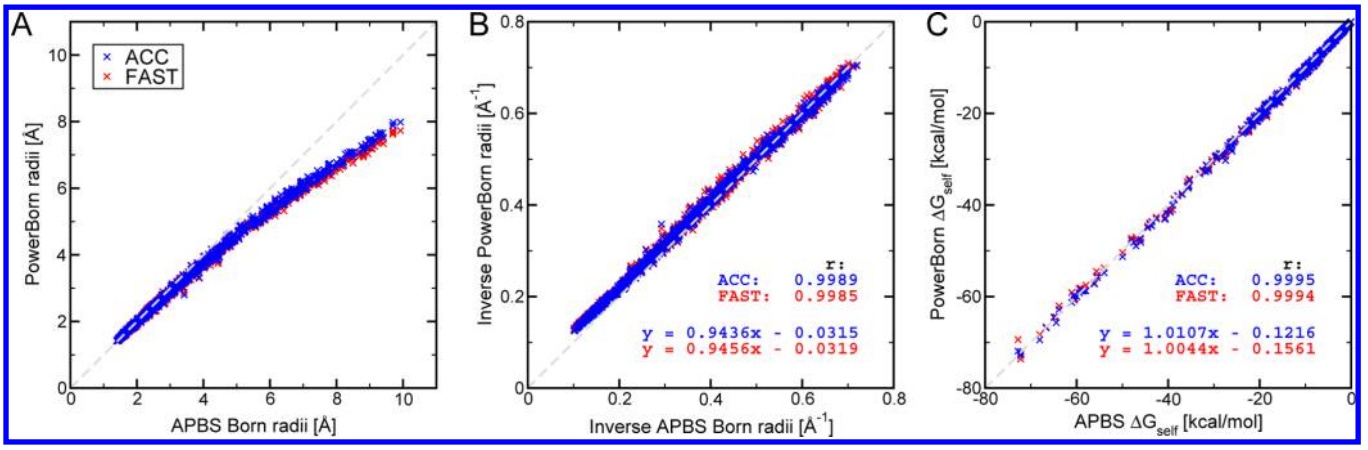


Figure 3. Comparison of Born radii (A), inverse Born radii (B), and self-polarization energies (C) for PDB 1SHG after applying the linear fit for the Born radii from eq 17.

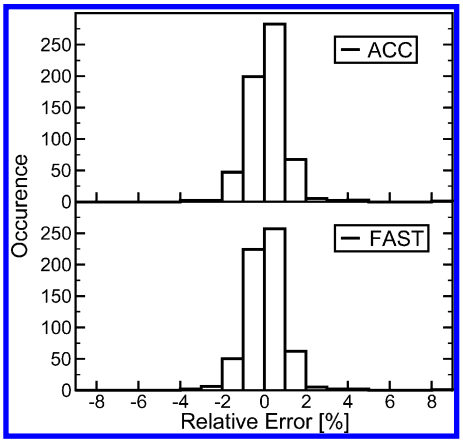


Figure 4. Histogram of relative errors between reference Poisson–Boltzmann computations using APBS[66] and generalized Born polar solvation free energies using PowerBorn radii for the 611 structures in the test set.

Table 2. Error Analysis for the Comparison of Polar Solvation Free Energies Computed by APBS and PowerBorn for the Considered Test Set of Protein Structures

parameter set	average unsigned error [%]	root-mean-square error [%]	maximal error [%]
ACC	0.66	0.94	8.18
FAST	0.68	0.96	8.16

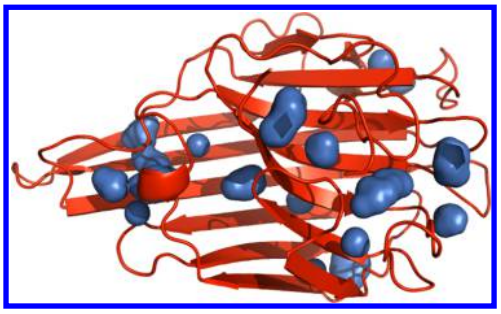


Figure 5. Cartoon representation (red) of the structure with the largest relative error in the test set (PDB 1NLS[70]) with cavities and tunnels in the solvent excluded surface (blue).

Another important aspect of this work is the power diagram representation used to construct the molecular surface. To estimate the accuracy impact of using the surface vertices provided by the power diagram, we have not taken them into account and instead increased the number of SAS sampling points on which water spheres are placed to yield at least the same number of water spheres as before. Recomputing the inverse Born radii for the PDB 1SHG shows that the correlation coefficient for the ACC method decreases from 0.9989 to 0.9964 as shown in Figure 6 and the standard deviation increases from 0.012 to 0.019 Å^{-1}.

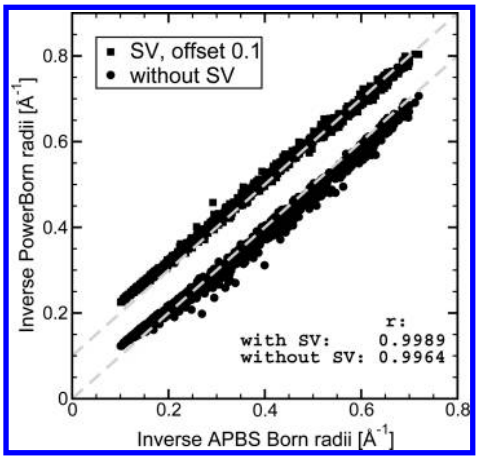


Figure 6. Comparison of the accuracy of inverse PowerBorn radii using ACC parameters to APBS[66] reference computations, if the solvent excluded surface is approximated with and without power diagram surface vertices (SV) from ref 63. Dashed gray lines mark perfect agreement.

Discretization Errors. As already noted when discussing the extensions for gradients to our algorithm, the grid based nature of our algorithm and the approximations made are likely to introduce discretization errors to the Born radii and polar solvation free energies, e.g., when comparing same conformations of a molecule that differ only by rigid rotations. Therefore, the computed polar solvation free energy may be split into an average value plus a contribution due to these errors

$$\Delta G = \Delta G_{\text{avg}} + \Delta G_{\text{discretization}} \tag{18}$$

This may be interpreted as our model having some additional degrees of freedom, e.g., three in the case of differing energies between rigidly rotated conformations, which have not yet been

integrated out. The sources of these discretization errors are the finite number of water spheres used to construct the SES approximation, the finite resolution of the octree, the decision to consider octree cells at the lowest level either completely inside or outside, but nothing in between, and the integration via the approximation in eq 7 in combination with the integration factor f from eq 6. To estimate the uncertainties that arise due to these approximations, we have performed 1000 random rigid rotations for each protein structure in the test set and calculated the standard deviation of the polar solvation free energy.

Figure 7 shows the resulting standard deviations with respect to the average computed value. We observe a correlation

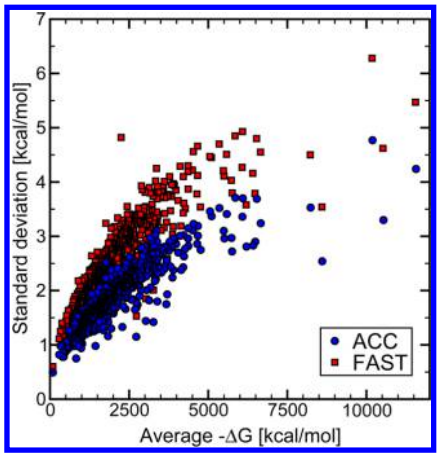


Figure 7. Standard deviations of the computed polar solvation free energies using PowerBorn radii after 1000 rigid rotations for each native protein structure in the test set.

between the size of the polar solvation free energy and the magnitude of the standard deviation, the latter growing approximately linear with the polar solvation free energy. While the difference of ACC and FAST in comparison to PB results was small, ACC features a much smaller discretization error, with a relative standard deviation averaged over the complete test set of 0.11% of the polar solvation free energies, whereas this value increases to 0.15% for FAST.

To set these numbers in relation, we note that the expected thermal fluctuation of the energy for a system with a Hamiltonian H is related to the heat capacity at constant pressure C_p of that system[71]

$$\langle H^2 \rangle - \langle H \rangle^2 = C_p R T^2 \tag{19}$$

Using ubiquitin as an example, the standard deviation of the polar solvation free energy for PDB 1UBQ is 2.25 kcal/mol for FAST, whereas the expected thermal fluctuation of the Hamiltonian using the heat capacity at constant pressure from Wintrode et. al[72] is on the order of 1490 kcal/mol at 298 K. Therefore, the discretization errors in the polar solvation free energy are much smaller than the expected thermal fluctuations of the system.

Nevertheless, the dependence of the fluctuations on the absolute values of the polar solvation free energy might cause changes in the free energy landscape. For example, the free energy difference between extended unfolded states that are highly solvated and have large negative polar solvation free energies and more compact folded states with smaller negative polar solvation free energies might be altered, due to the larger fluctuations in the unfolded state. This may be compensated by using a set of increased atomic radii. These will induce a larger penalty on extended than compact states, since in the latter case most atoms are deeply buried in the core anyway and therefore not affected by the increased atomic radii. Thus, we believe that the discretization errors do not pose a significant problem to the application of this algorithm to molecular Monte Carlo simulations and can be overcome with careful chosen force field parameters.

Performance Measurements. Since our goal was not only to provide yet another accurate method for computing Born radii, but also a computationally very efficient one, we compare the speed of our algorithm to two other state of the art methods. The GBOBC method[46] as implemented in GROMACS[73] using optimized kernels is one of the most common and fastest methods available, but being based on the Coulomb-field approximation, it is not the most accurate. On the other hand, we use the GBMV2[49] method as implemented in CHARMM[74] as a benchmark for a method with the same level of accuracy as our algorithm. For this test, we have chosen a smaller set of native PDB structures listed in the Supporting Information. For further details on the timing and speedup measurements, please refer to the Appendix.

The absolute time for one PowerBorn evaluation depending on the protein structure size is illustrated in Figure 8. We find

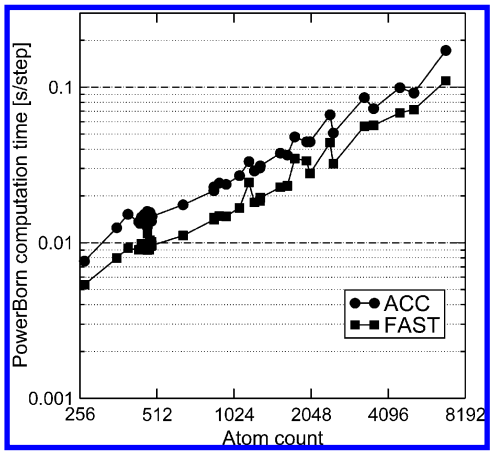


Figure 8. Computation time of one PowerBorn evaluation as a function of the number of atoms in the protein.

that our method has a reasonably small overhead with evaluation times ranging from 5.4 ms using FAST for the smallest protein in the timing test set and reaching up to 0.11 s using ACC for the largest protein. The speedup measurements in Figure 9 show that the ACC version is at least 4.16 times faster than GBMV2, while the FAST version yields a factor of 5.96 for very small molecules. The speedup increases up to 10.74 and 14.17 respectively for larger proteins, demonstrating that our algorithm significantly outperforms GBMV2 while delivering the same level of accuracy.

However, for small molecules the GBOBC method outperforms our algorithm clearly, being up to a factor of 5.3 times faster when compared to FAST. The performance of ACC breaks even with that of GBOBC when a structure size of approximately 2000 atoms is reached, while the break-even point for FAST is at roughly 1250 atoms. Increasing the structure size, the advantage of our algorithm becomes more

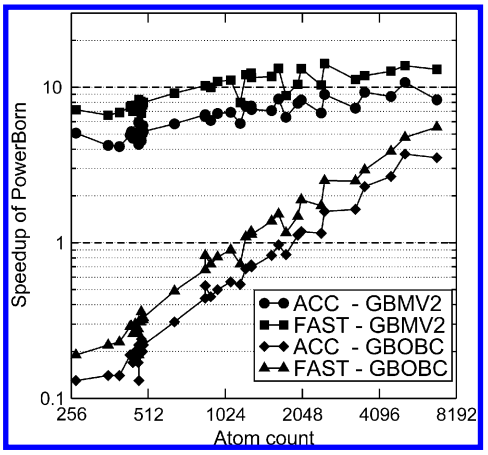


Figure 9. Speedup of PowerBorn's ACC and FAST version in comparison to GBMV2[49] in CHARMM[74] and GBOBC[46] in GROMACS[73] for different sized protein structures in the timing test set.

obvious with speedups reaching 3.72 for ACC and 5.53 for FAST.

CONCLUSION

We have presented our new PowerBorn algorithm to compute Born radii for use in the generalized Born model, a common ingredient in many implicit solvent models used in biomolecular simulations, based on a Barnes–Hut tree code scheme. To evaluate the integral expression for the Born radii, we construct an octree representation of the molecule and its surrounding water and perform top-down tree-walks in the integration step to exploit the rapid decay of the integrand. Comparing the accuracy in relation to Poisson–Boltzmann results, we find very good agreement for computed Born radii with correlation coefficients larger than 0.9978 and root-mean-square errors for polar solvation free energies close to 1%. For the largest disagreement found in the test set, closer examination showed that this can be attributed to a known deficiency of the GB model and not our implementation. The investigation of the discretization errors implied by the grid based nature of our algorithm and the used approximations showed that these are much smaller than the expected thermal fluctuations of the energy for a typical system and that their influence on the free energy landscape is likely to be compensated by careful chosen force-field parameters. We also demonstrated the superior numerical performance of our new algorithm, reaching a speedup factor of up to 14.17 in comparison to the equally accurate GBMV2 method in CHARMM and 5.53 in comparison to the GBOBC method in GROMACS for large protein structures, while for very small peptides, the latter still outperforms our algorithm. Our C++ implementation of this algorithm will be available at http://www.int.kit.edu/nanosim/ and is also part of the SIMONA simulation package.[60]

APPENDIX

Program Input Preparation

All PQR molecule files used with PowerBorn or APBS were prepared using the Ambertools tleap program[75] employing mBondi2 atomic radii.[46] The probe radius for the solvent excluded surface was 1.4 Å. Dielectric constants of protein and water were 1.0 and 80.0, respectively. APBS computations were done with a finest grid spacing of 0.25 Å, multiple Debye–Hückel boundary conditions, and a molecular surface with no smoothing. CHARMM[74] input was generated using CHARMM-GUI.[76] Input files for GROMACS[73] were prepared with pdb2gmx using the Amber99SB[77] force field and corresponding implicit solvent radii. All images of protein structures were created with PyMOL.[78]

Timing Measurements

GBMV2 timings were performed using CHARMM (version 35b5) with the FAST method enabled. Input parameters were changed to those given in ref 49, except the radial grid, which had to be kept at the default values to avoid segmentation faults when running CHARMM. Surface area computation was disabled. For timing measurements, a 100 step energy minimization was run for each structure, and the timings were collected from the CHARMM internal timer "GB_RADII." We also investigated the code related to that timer and found that it only measures the time for computing the Born radii and not the gradients. However, it caches already computed values and neighbor lists for later use in the force computation. Thus, the timings should yield a reliable comparison to our PowerBorn method, which does not compute forces.

GBOBC timings were performed using GROMACS (version 4.5.1) with optimized kernels. The input structures were minimized, and afterwards a 1000 step MD simulation was run using infinite cutoffs during which the time of the Born radii computation (C function calc_gb_rad) was measured using C's clock function. The GROMACS function does also compute some components of the derivatives of the Born radii. Since the employed analytic pairwise integration scheme (equations 13–15 in ref 25) depends only on the distance between two atoms, computation of the derivatives of the Born radii integral is unlikely to add significant computational load, since most terms that contribute to the derivative are also needed for the Born radii. Furthermore, the computationally expensive logarithmic term in eq 13 in ref 25 is not required for the computation of the derivatives. Therefore, we believe that these timings are also comparable to timings of our PowerBorn method.

PowerBorn timings were measured by recomputing all Born radii 1000 times for the same native input structure. Time was again measured with C's clock function. All timing computations are done on a single core of a 2.4 GHz Intel Wolfdale CPU with 4 GB of RAM running OpenSuse 11.4. The GCC compiler suite 4.5.1 was used.

ASSOCIATED CONTENT

Supporting Information

List of PDB IDs used for timing measurements. Full expression for the analytic integral outside the bounding box and visualization of the surface around the three atoms with the largest Born radii deviation for the 1UBQ test case. This material is available free of charge via the Internet at http://pubs.acs.org.

AUTHOR INFORMATION

Corresponding Author

*E-mail: wolfgang.wenzel@kit.edu.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors would like to thank Ivan Kondov, Julia Setzler, and Timo Strunk for helpful discussion on the manuscript. This work was performed under a grant of the HPC program of the Baden-Württemberg Stiftung (HPC-5). M.B. acknowledges the support from Simulation Lab NanoMikro at Steinbuch Centre for Computing and the stipend of the Landesgraduiertenförderung Baden-Württemberg.

■ REFERENCES

(1) Dill, K. A.; Truskett, T. M.; Vlachy, V.; Hribar-Lee, B. Modeling Water, the Hydrophobic Effect, and Ion Solvation. *Annu. Rev. Biophys. Biomol. Struct.* **2005**, *34*, 173−199.
(2) Hu, H.; Yang, W. Free Energies of Chemical Reactions in Solution and in Enzymes with Ab Initio Quantum Mechanics/Molecular Mechanics Methods. *Annu. Rev. Phys. Chem.* **2008**, *59*, 573−601.
(3) Roux, B.; Simonson, T. Implicit solvent models. *Biophys. Chem.* **1999**, *78*, 1−20.
(4) Levy, R. M.; Gallicchio, E. Computer Simulations with Explicit Solvent: Recent Progress in the Thermodynamic Decomposition of Free Energies and in Modeling Electrostatic Effects. *Annu. Rev. Phys. Chem.* **1998**, *49*, 531−567.
(5) Tomasi, J.; Persico, M. Molecular Interactions in Solution: An Overview of Methods Based on Continuous Distributions of the Solvent. *Chem. Rev.* **1994**, *94*, 2027−2094.
(6) Tomasi, J.; Mennucci, B.; Cammi, R. Quantum Mechanical Continuum Solvation Models. *Chem. Rev.* **2005**, *105*, 2999−3094.
(7) Orozco, M.; Luque, F. J. Theoretical Methods for the Description of the Solvent Effect in Biomolecular Systems. *Chem. Rev.* **2000**, *100*, 4187−4226.
(8) Cramer, C. J.; Truhlar, D. G. Implicit solvation models: Equilibria, structure, spectra, and dynamics. *Chem. Rev.* **1999**, *99*, 2161−2200.
(9) Feig, M.; Brooks, C. L. Recent advances in the development and application of implicit solvent models in biomolecule simulations. *Curr. Opin. Struct. Biol.* **2004**, *14*, 217−224.
(10) Baker, N. A.; Bashford, D.; Case, D. A. Implicit Solvent Electrostatics in Biomolecular Simulation. In *New Algorithms for Macromolecular Simulation*; Leimkuhler, B., Chipot, C., Elber, R., Laaksonen, A., Mark, A., Schlick, T., Schütte, C., Skeel, R., Barth, T. J., Griebel, M., Keyes, D. E., Nieminen, R. M., Roose, D., Schlick, T., Eds.; Lecture Notes in Computational Science and Engineering; Springer: Berlin, 2006; Vol. *49*, pp 263−295.
(11) Grochowski, P.; Trylska, J. Continuum molecular electrostatics, salt effects, and counterion binding—A review of the Poisson−Boltzmann theory and its modifications. *Biopolymers* **2008**, *89*, 93−113.
(12) Chen, J.; Brooks, C. L.; Khandogin, J. Recent advances in implicit solvent-based methods for biomolecular simulations. *Curr. Opin. Struct. Biol.* **2008**, *18*, 140−148.
(13) Honig, B.; Nicholls, A. Classical electrostatics in biology and chemistry. *Science* **1995**, *268*, 1144−1149.
(14) Beroza, P.; Case, D. A. Calculations of proton-binding thermodynamics in proteins. *Methods Enzymol.* **1998**, *295*, 170−189.
(15) Scarsi, M.; Apostolakis, J.; Caflisch, A. Continuum electrostatic energies of macromolecules in aqueous solutions. *J. Phys. Chem. A* **1997**, *101*, 8098−8106.
(16) Chin, K.; Sharp, K. A.; Honig, B.; Pyle, A. M. Calculating the electrostatic properties of RNA provides new insights into molecular interactions and function. *Nat. Struct. Biol.* **1999**, *6*, 1055−1061.
(17) Luo, R.; David, L.; Gilson, M. K. Accelerated Poisson-Boltzmann calculations for static and dynamic systems. *J. Comput. Chem.* **2002**, *23*, 1244−1253.
(18) Mackerell, A. D. Empirical force fields for biological macromolecules: Overview and issues. *J. Comput. Chem.* **2004**, *25*, 1584−1604.
(19) Baker, N. A. Biomolecular Applications of Poisson−Boltzmann Methods. In *Reviews in Computational Chemistry*; Lipkowitz, K. B., Larter, R., Cundari, T. R., Eds.; John Wiley & Sons, Inc.: New York, 2005; pp 349−379.
(20) Baker, N. A. Improving implicit solvent simulations: a Poisson-centric view. *Curr. Opin. Struct. Biol.* **2005**, *15*, 137−143.
(21) Madura, J. D.; Briggs, J. M.; Wade, R. C.; Davis, M. E.; Luty, B. A.; Ilin, A.; Antosiewicz, J.; Gilson, M. K.; Bagheri, B.; Scott, L. R.; McCammon, J. A. Electrostatics and diffusion of molecules in solution: simulations with the University of Houston Brownian Dynamics program. *Comput. Phys. Commun.* **1995**, *91*, 57−95.
(22) Gilson, M. K. Theory of electrostatic interactions in macromolecules. *Curr. Opin. Struct. Biol.* **1995**, *5*, 216−223.
(23) Born, M. Volumen und Hydratationswärme der Ionen. *Z. Phys.* **1920**, *1*, 45−48.
(24) Bashford, D.; Case, D. A. Generalized born models of macromolecular solvation effects. *Annu. Rev. Phys. Chem.* **2000**, *51*, 129−152.
(25) Hawkins, G.; Cramer, C.; Truhlar, D. Pairwise solute descreening of solute charges from a dielectric medium. *Chem. Phys. Lett.* **1995**, *246*, 122−129.
(26) Schaefer, M.; Karplus, M. A comprehensive analytical treatment of continuum electrostatics. *J. Phys. Chem.* **1996**, *100*, 1578−1599.
(27) Qiu, D.; Shenkin, P. S.; Hollinger, F. P.; Still, W. C. The GB/SA Continuum Model for Solvation. A Fast Analytical Method for the Calculation of Approximate Born Radii. *J. Phys. Chem. A* **1997**, *101*, 3005−3014.
(28) Lee, M. S.; Salsbury, F. R.; Brooks, C. L. Novel generalized Born methods. *J. Chem. Phys.* **2002**, *116*, 10606.
(29) Romanov, A. N.; Jabin, S. N.; Martynov, Y. B.; Sulimov, A. V.; Grigoriev, F. V.; Sulimov, V. B. Surface generalized born method: A simple, fast, and precise implicit solvent model beyond the coulomb approximation. *J. Phys. Chem. A* **2004**, *108*, 9323−9327.
(30) Tjong, H.; Zhou, H.-X. GBr6: A Parameterization-Free, Accurate, Analytical Generalized Born Method. *J. Phys. Chem. B* **2007**, *111*, 3055−3061.
(31) Gallicchio, E.; Levy, R. M. AGBNP: An analytic implicit solvent model suitable for molecular dynamics simulations and high-resolution modeling. *J. Comput. Chem.* **2004**, *25*, 479−499.
(32) Still, W. C.; Tempczyk, A.; Hawley, R. C.; Hendrickson, T. Semianalytical treatment of solvation for molecular mechanics and dynamics. *J. Am. Chem. Soc.* **1990**, *112*, 6127−6129.
(33) Hawkins, G. D.; Cramer, C. J.; Truhlar, D. G. Parametrized Models of Aqueous Free Energies of Solvation Based on Pairwise Descreening of Solute Atomic Charges from a Dielectric Medium. *J. Phys. Chem.* **1996**, *100*, 19824−19839.
(34) Ghosh, A.; Rapp, C. S.; Friesner, R. A. Generalized born model based on a surface integral formulation. *J. Phys. Chem. B* **1998**, *102*, 10983−10990.
(35) Calimet, N.; Schaefer, M.; Simonson, T. Protein molecular dynamics with the generalized born/ACE solvent model. *Proteins: Struct., Funct., Bioinf.* **2001**, *45*, 144−158.
(36) Fan, H.; Mark, A. E.; Zhu, J.; Honig, B. Comparative study of generalized Born models: Protein dynamics. *Proc. Natl. Acad. Sci. U. S. A.* **2005**, *102*, 6760−6764.
(37) Dominy, B. N.; Brooks, C. L. Development of a generalized born model parametrization for proteins and nucleic acids. *J. Phys. Chem. B* **1999**, *103*, 3765−3773.
(38) Tsui, V.; Case, D. A. Molecular dynamics simulations of nucleic acids with a generalized born solvation model. *J. Am. Chem. Soc.* **2000**, *122*, 2489−2498.
(39) Wang, T.; Wade, R. C. Implicit solvent models for flexible protein-protein docking by molecular dynamics simulation. *Proteins: Struct., Funct., Genet.* **2003**, *50*, 158−169.
(40) Simmerling, C.; Strockbine, B.; Roitberg, A. E. All-atom structure prediction and folding simulations of a stable protein. *J. Am. Chem. Soc.* **2002**, *124*, 11258−11259.
(41) Richards, F. M. Areas, Volumes, Packing, and Protein Structure. *Annu. Rev. Biomed. Eng.* **1977**, *6*, 151−176.

(42) Connolly, M. Solvent-accessible surfaces of proteins and nucleic-acids. *Science* **1983**, *221*, 709−713.
(43) Connolly, M. Analytical Molecular-Surface Calculation. *J. Appl. Crystallogr.* **1983**, *16*, 548−558.
(44) Lee, M. S.; Olson, M. A. Evaluation of Poisson Solvation Models Using a Hybrid Explicit/Implicit Solvent Method. *J. Phys. Chem. B* **2005**, *109*, 5223−5236.
(45) Wagoner, J.; Baker, N. A. Solvation forces on biomolecular structures: A comparison of explicit solvent and Poisson−Boltzmann models. *J. Comput. Chem.* **2004**, *25*, 1623−1629.
(46) Onufriev, A.; Bashford, D.; Case, D. A. Exploring protein native states and large-scale conformational changes with a modified generalized born model. *Proteins: Struct., Funct., Bioinf.* **2004**, *55*, 383−394.
(47) Mongan, J.; Simmerling, C.; McCammon, J. A.; Case, D. A.; Onufriev, A. Generalized Born Model with a Simple, Robust Molecular Volume Correction. *J. Chem. Theory Comput.* **2007**, *3*, 156−169.
(48) Gallicchio, E.; Paris, K.; Levy, R. M. The AGBNP2 Implicit Solvation Model. *J. Chem. Theory Comput.* **2009**, *5*, 2544−2564.
(49) Lee, M. S.; Feig, M.; Salsbury, F. R., Jr; Brooks, C. L., 3rd New analytic approximation to the standard molecular volume definition and its application to generalized Born calculations. *J. Comput. Chem.* **2003**, *24*, 1348−1356.
(50) Lee, M. S.; Salsbury, F. R.; Brooks, C. L. Novel generalized Born methods. *J. Chem. Phys.* **2002**, *116*, 10606.
(51) Im, W.; Lee, M. S.; Brooks, C. L., III Generalized born model with a simple smoothing function. *J. Comput. Chem.* **2003**, *24*, 1691−1702.
(52) Gallicchio, E.; Zhang, L. Y.; Levy, R. M. The SGB/NP hydration free energy model based on the surface generalized born solvent reaction field and novel nonpolar hydration free energy estimators. *J. Comput. Chem.* **2002**, *23*, 517−529.
(53) Ghosh, A.; Rapp, C. S.; Friesner, R. A. Generalized born model based on a surface integral formulation. *J. Phys. Chem. B* **1998**, *102*, 10983−10990.
(54) Kirkwood, J. G. Theory of Solutions of Molecules Containing Widely Separated Charges with Special Application to Zwitterions. *J. Chem. Phys.* **1934**, *2*, 351−361.
(55) Grycuk, T. Deficiency of the Coulomb-field approximation in the generalized Born model: An improved formula for Born radii evaluation. *J. Chem. Phys.* **2003**, *119*, 4817−4826.
(56) Mongan, J.; Svrcek-Seiler, W. A.; Onufriev, A. Analysis of integral expressions for effective Born radii. *J. Chem. Phys.* **2007**, *127*, 185101.
(57) Aguilar, B.; Shadrach, R.; Onufriev, A. V. Reducing the Secondary Structure Bias in the Generalized Born Model via R6 Effective Radii. *J. Chem. Theory Comput.* **2010**, *6*, 3613−3630.
(58) Zhu, J.; Alexov, E.; Honig, B. Comparative study of generalized Born models: Born radii and peptide folding. *J. Phys. Chem. B* **2005**, *109*, 3008−3022.
(59) Onufriev, A. V.; Sigalov, G. A strategy for reducing gross errors in the generalized Born models of implicit solvation. *J. Chem. Phys.* **2011**, *134*, 164104.
(60) Strunk, T.; Wolf, M.; Brieg, M.; Klenin, K.; Biewer, A.; Tristram, F.; Ernst, M.; Kleine, P. J.; Heilmann, N.; Kondov, I.; Wenzel, W. SIMONA 1.0: An efficient and versatile framework for stochastic simulations of molecular and nanoscale systems. *J. Comput. Chem.* **2012**, *33*, 2602.
(61) Barnes, J.; Hut, P. A hierarchical O(N log N) force-calculation algorithm. *Nature* **1986**, *324*, 446−449.
(62) Lee, B.; Richards, F. M. The interpretation of protein structures: Estimation of static accessibility. *J. Mol. Biol.* **1971**, *55*, 379−400.
(63) Klenin, K. V.; Tristram, F.; Strunk, T.; Wenzel, W. Derivatives of molecular surface area and volume: Simple and exact analytical formulas. *J. Comput. Chem.* **2011**, *32*, 2647−2653.
(64) Musacchio, A.; Noble, M.; Pauptit, R.; Wierenga, R.; Saraste, M. Crystal structure of a Src-homology 3 (SH3) domain. *Nature* **1992**, *359*, 851−855.
(65) Chocholoušová, J.; Feig, M. Balancing an accurate representation of the molecular surface in generalized born formalisms with integrator stability in molecular dynamics simulations. *J. Comput. Chem.* **2006**, *27*, 719−729.
(66) Baker, N. A.; Sept, D.; Joseph, S.; Holst, M. J.; McCammon, J. A. Electrostatics of nanosystems: Application to microtubules and the ribosome. *Proc. Natl. Acad. Sci. U. S. A.* **2001**, *98*, 10037−10041.
(67) Mattinen, M.-L.; Kontteli, M.; Kerovuo, J.; Drakenberg, T.; Annila, A.; Linder, M.; Reinikainen, T.; Lindeberg, G. Three-dimensional structures of three engineered cellulose-binding domains of cellobiohydrolase I from Trichoderma reesei. *Protein Sci.* **1997**, *6*, 294−303.
(68) Vijay-Kumar, S.; Bugg, C. E.; Cook, W. J. Structure of ubiquitin refined at 1.8 Å resolution. *J. Mol. Biol.* **1987**, *194*, 531−544.
(69) Feig, M.; Onufriev, A.; Lee, M. S.; Im, W.; Case, D. A.; Brooks, C. L., III. Performance comparison of generalized born and Poisson methods in the calculation of electrostatic solvation energies for protein structures. *J. Comput. Chem.* **2004**, *25*, 265−284.
(70) Deacon, A.; Gleichmann, T.; Kalb (Gilboa), A. J.; Price, H.; Raftery, J.; Bradbrook, G.; Yariv, J.; Helliwell, J. R. The structure of concanavalin A and its bound solvent determined with small-molecule accuracy at 0.94 [Aring] resolution. *J. Chem. Soc., Faraday Trans.* **1997**, *93*, 4305−4312.
(71) Gómez, J.; Hilser, V. J.; Xie, D.; Freire, E. The heat capacity of proteins. *Proteins: Struct., Funct., Bioinf.* **1995**, *22*, 404−412.
(72) Wintrode, P. L.; Makhatadze, G. I.; Privalov, P. L. Thermodynamics of ubiquitin unfolding. *Proteins: Struct., Funct., Bioinf.* **1994**, *18*, 246−253.
(73) Hess, B.; Kutzner, C.; Van der Spoel, D.; Lindahl, E. GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation. *J. Chem. Theory Comput.* **2008**, *4*, 435−447.
(74) Brooks, B. R.; Brooks, C. L.; Mackerell, A. D.; Nilsson, L.; Petrella, R. J.; Roux, B.; Won, Y.; Archontis, G.; Bartels, C.; Boresch, S.; Caflisch, A.; Caves, L.; Cui, Q.; Dinner, A. R.; Feig, M.; Fischer, S.; Gao, J.; Hodoscek, M.; Im, W.; Kuczera, K.; Lazaridis, T.; Ma, J.; Ovchinnikov, V.; Paci, E.; Pastor, R. W.; Post, C. B.; Pu, J. Z.; Schaefer, M.; Tidor, B.; Venable, R. M.; Woodcock, H. L.; Wu, X.; Yang, W.; York, D. M.; Karplus, M. CHARMM: The Biomolecular Simulation Program. *J. Comput. Chem.* **2009**, *30*, 1545−1614.
(75) Case, D. A.; Cheatham, T. E.; Darden, T.; Gohlke, H.; Luo, R.; Merz, K. M.; Onufriev, A.; Simmerling, C.; Wang, B.; Woods, R. J. The Amber biomolecular simulation programs. *J. Comput. Chem.* **2005**, *26*, 1668−1688.
(76) Jo, S.; Kim, T.; Iyer, V. G.; Im, W. CHARMM-GUI: A web-based graphical user interface for CHARMM. *J. Comput. Chem.* **2008**, *29*, 1859−1865.
(77) Hornak, V.; Abel, R.; Okur, A.; Strockbine, B.; Roitberg, A.; Simmerling, C. Comparison of multiple Amber force fields and development of improved protein backbone parameters. *Proteins: Struct., Funct., Bioinf.* **2006**, *65*, 712−725.
(78) *The PyMOL Molecular Graphics System*, version 1.3r1; Schrödinger, LLC: New York, 2010.