Maxim Totrov
Ruben Abagyan
Molsoft LLC,
3366 North Torrey Pines Ct.,
San Diego, CA 92037

Rapid Boundary Element Solvation Electrostatics Calculations in Folding Simulations: Successful Folding of a 23-Residue Peptide

Abstract: *Solvation effects play a profound role in the energetics of protein folding. While a continuum dielectric model of solvation may provide a sufficiently accurate estimate of the solvation effects, until now this model was too computationally expensive and unstable for folding simulations. Here we proposed a fast yet accurate and robust implementation of the boundary element solution of the Poisson equation, the REBEL algorithm. Using our earlier double-energy scheme, we included for the first time the mathematically rigorous continuous REBEL solvation term in our Biased Probability Monte Carlo (BPMC) simulations of the peptide folding. The free energy of a 23-residue ββα-peptide was then globally optimized with and without the solvation electrostatics contribution. An ensemble of ββα conformations was found at and near the global minimum of the energy function with the REBEL electrostatic solvation term. Much poorer correspondence to the native solution structure was found in the "control" simulations with a traditional method to account for solvation via a distance-dependent dielectric constant. Each simulation took less than 40 h (21 h without electrostatic solvation calculation) on a single Alpha 677 MHz CPU and involved more than 40,000 solvation energy evaluations. This work demonstrates for the first time that such a simulation can be performed in a realistic time frame. The proposed procedure may eliminate the energy evaluation accuracy bottleneck in folding simulations.* © 2001 John Wiley & Sons, Inc. Biopolymers (Pept Sci) 60: 124–133, 2001

Keywords: *electrostatics; protein folding; computer simulations; structure prediction; solvation; hydration energy; boundary element; Poisson equation; REBEL*

INTRODUCTION

It is well recognized that electrostatic interactions have profound effects on macromolecular structure, folding, and binding. Simplistic pairwise Coulomb energy used in a number of molecular force fields proved to be inadequate in many cases since it does not account for the solvent effects. Presence of a highly polar solvent (water) effectively screens the interaction of the charges of the solute, and favors the exposure of the charged and polar groups on the surface. The most rigorous treatment of solvation effects might be the inclusion of explicit water molecules into the system. Such calculations require addition of thousands of new atoms for a moderately sized macromolecule, and a long run of molecular

Correspondence to: Ruben Abagyan, The Scripps Research Institute, MB37 10550 North Torrey Pines Rd., San Diego, CA 92037

Biopolymers (Peptide Science), Vol. 60, 124–133 (2001)

dynamics is necessary to achieve even rather superficial sampling of the phase space for the added water molecules. Extensive sampling is needed since the solvent molecules are not static and their thermal motion is essential for the electrostatic properties of the solvent.[1] Furthermore, the energy accuracy in these simulations is limited by simplifications of nonpolarizable explicit water models. Modeling of protein folding with explicit water thus remains limited to short peptides.[2]

As an alternative to explicit solvent, one can use the continuous dielectric model that has recently been successfully used to evaluate thermodynamic stability of conformational ensembles.[3–5] Instead of the discrete molecules, the solvent is represented as a continuum of high dielectric constant. In this case, the electrostatic energy may be calculated as the energy of a set of point charges in a low dielectric constant medium surrounded by a high dielectric constant medium. The molecular surface is usually taken as a boundary between the two. Unfortunately, analytic solutions exist only for special shapes, e.g., a spherical boundary. Certain methods utilize these analytic solutions to obtain relatively simple approximations of energy under an assumption that the protein has quasi-spherical shape, e.g., the image charge approximation.[6−8] Generalized Born methods use similar assumptions (reviewed by Bashford and Case[9]). The precision of these approximations is limited. A more rigorous approach is numeric solution of the differential Poisson equation

$$-\nabla(\varepsilon(\mathbf{r})\nabla\phi(\mathbf{r})) = \rho(\mathbf{r})$$

where ε is the dielectric constant (permittivity), ϕ is the electric potential, and ρ is the charge density. Two major approaches have been developed: the finite difference method[10] and the boundary element method.[11,12]

The finite difference method is currently the most popular approach. The major disadvantage of the method is that it requires manipulations of very large three-dimensional arrays since the properties such as electrostatic potential, charges, and dielectric constant have to be represented on a three-dimensional grid. To achieve an adequate precision, the sub-Ångstrom grids are required. For a medium-sized system of 50 Å diameter and grid step of 0.5 Å, the calculations involve the manipulation of several arrays of 100 × 100 × 100 = one million values each, which is both slow and memory-consuming. Another shortcoming of the finite difference method is the strong dependency of the results on the grid step, size, and position, since both the dielectric boundary and the point charges are projected to a grid instead of being kept at their original positions.

The boundary element method is based on the following mathematical observation: the solution of Poisson equation for the system where the space is divided into two regions of different dielectric permittivity can be represented as the solution for a uniform medium if certain additional electrical charge is distributed over the boundary between the regions. Since the electric field in the uniform medium obeys the Coulomb law, once the charge density on the boundary is known, electrostatic potentials and energy can be easily calculated. However, to find the boundary charge distribution, an integral equation has to be solved. The efficiency of the method depends to a great extent on the implementation of this solution.

Discretization of the molecular surface is an important part of the procedure, and has a crucial effect on its performance and precision. The number of surface elements needs to be kept as small as possible to make the calculations fast and to reduce the amount of memory needed. On the other hand, the shape of the surface has to be adequately represented in order to achieve good precision. It is hard to satisfy both of these conflicting requirements. Usually the surface is divided into triangles used as boundary elements. One of the approaches essentially projects a pretriangulated sphere onto the molecular surface.[12] While producing a relatively small number of surface elements, it will only generate satisfactory surface representation for a quasi-spherical molecule and any clefts, which are quite common for enzymes and receptors, may get severely distorted. Another approach is to use the detailed triangulated molecular surface,[11] which guarantees good precision but is extremely slow and requires large memory to store the matrix. The fast multipole approximation was used by Nicholls and colleagues[13] to accelerate the boundary element method calculations and was shown to be efficient for boundaries comprising more than 10,000 elements. Up to 30-fold acceleration was achieved for 40,000 boundary elements. In the present work we take a different approach, to avoid both the excessive number of boundary elements and the oversimplification of the surface. We show that it is possible to reduce the number of boundary elements more than 10-fold without significant impact on the precision. Instead of directly using triangles as boundary elements, we combine them into relatively few composite patches of arbitrary shape. The limited number of boundary elements keeps the calculations fast while the accurate shape of the elements, to a large extent, preserves the precision. We found that this approach, while simpler than the multigrid boundary element proposed by Vorobiev and Scheraga,[14] is much faster and sufficiently precise.

Folding simulations with electrostatic solvation term were so far limited to rather short fragments with little or no stable secondary structure.[15] Here we report folding simulations of a 23-residue peptide Acet–Tyr–Thr–Val–D–Pro–Ser–Fen–Thr–Phe–Ser–Arg–Ser–Asp–Glu–Leu–Ala–Lys–Leu–Leu–Arg–Leu–His–Ala–Gly–$CONH_2$, designed to adopt a zinc-finger-like $\beta\beta\alpha$ fold without a metal ion. The “Fen” residue in the sequence of the peptide is an unnatural amino acid, 3-(1,10-phenanthrol-2-yl)-L-alanine. The solution structure of this peptide has been solved by NMR[16] and contains both basic types of secondary structure elements observed in larger proteins.

THE BOUNDARY ELEMENT METHOD

The basic task in continuum dielectric solvation electrostatics calculation is to find the electric potential or field produced by a system of charges q_i in the region of space with dielectric constant ε_{in} surrounded by a medium with dielectric constant ε_{out}. The idea of the boundary element method is to find a charge distribution on the dielectric boundary surface that reproduces the electric field in the uniform medium with dielectric constant ε_{in}. Once such a distribution is found, one can calculate the electrostatic potential at any point from the Coulomb law since the dielectric medium is now uniform.

Theory

The electric field at an arbitrary point on the boundary should obey two conditions that can be used to deduce an equation for the surface charge density σ. The first condition is the continuity of the normal component of the electric displacement vector at any point on the boundary. If $\mathbf{n}$ is the normal to the boundary, $\mathbf{D}_{\text{in}}$ is the displacement just inside the boundary and $\mathbf{D}_{\text{out}}$ is the displacement just outside the boundary, than

$$\mathbf{D}_{\text{in}} \cdot \mathbf{n} = \mathbf{D}_{\text{out}} \cdot \mathbf{n} \tag{1}$$

The second condition is for the discontinuity of the normal component of the electric field:

$$(\mathbf{E}_{\text{out}} - \mathbf{E}_{\text{in}}) \cdot \mathbf{n} = 4\pi\sigma \tag{2}$$

where $\mathbf{E}_{\text{out}}$ and $\mathbf{E}_{\text{in}}$ are the electric field vectors just outside and just inside the boundary, respectively. Combining these two equations and taking into account that $\mathbf{D} = \varepsilon\mathbf{E}$ we obtain the following equation relating $\mathbf{E}_{\text{out}}$ and σ:

$$\sigma = \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{4\pi\varepsilon_{\text{in}}}\right)\mathbf{E}_{\text{out}} \cdot \mathbf{n} \tag{3}$$

On the other hand, the electric field $\mathbf{E}$ can be calculated with the help of Coulomb’s law from the known electric charges q_i and the charge density distribution σ:

$$\mathbf{E} = \sum_i \frac{q_i(\mathbf{r} - \mathbf{r}_i)}{\varepsilon_{\text{in}}|\mathbf{r} - \mathbf{r}_i|^3} + \oiint \frac{(\mathbf{r} - \mathbf{r}_s)}{|\mathbf{r} - \mathbf{r}_s|^3}\,\sigma_s ds \tag{4}$$

where $\mathbf{r}$ is the radius-vector of the point where the electric field is being calculated, $\mathbf{r}_i$ is the radius vectors of the charges q_i, r_s, and σ_s are the radius-vector and the surface charge density of an infinitesimal element of the boundary ds and the integral is taken over the whole boundary. This expression should not, however, be directly used in Eq. (3), because the surface integral in Eq. (4) has a discontinuity at each surface point. It can be shown that the value of the integral at the surface point and at the point infinitely close to it but outside the surface differ by $2\pi\sigma\mathbf{n}$, and for $\mathbf{E}_{\text{out}}$ one can have

$$\mathbf{E}_{\text{out}} = \sum_i \frac{q_i(\mathbf{r} - \mathbf{r}_i)}{\varepsilon_{\text{in}}|\mathbf{r} - \mathbf{r}_i|^3} + 2\pi\sigma\mathbf{n} + \oiint \frac{(\mathbf{r} - \mathbf{r}_s)}{|\mathbf{r} - \mathbf{r}_s|^3}\,\sigma_s ds \tag{5}$$

where $\mathbf{r}$ is now the radius-vector of a point on the boundary.

Now we can substitute $\mathbf{E}_{\text{out}}$ in Eq. (3) and obtain an integral equation for the σ:

$$\sigma = \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{4\pi\varepsilon_{\text{in}}^2}\right)\sum_i \frac{q_i(\mathbf{r} - \mathbf{r}_i)\cdot\mathbf{n}}{|\mathbf{r} - \mathbf{r}_i|^3} + \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\varepsilon_{\text{in}}}\right)\sigma + \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{4\pi\varepsilon_{\text{in}}}\right)\oiint \frac{\sigma_s(\mathbf{r} - \mathbf{r}_s)\cdot\mathbf{n}}{|\mathbf{r} - \mathbf{r}_s|^3}\,ds \tag{6}$$

or

$$\sigma - \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})}\right)\oiint \frac{\sigma_s(\mathbf{r} - \mathbf{r}_s)\cdot\mathbf{n}}{|\mathbf{r} - \mathbf{r}_s|^3}\,ds = \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})}\right)\frac{1}{\varepsilon_{in}}\sum_i \frac{q_i(\mathbf{r} - \mathbf{r}_i)\cdot\mathbf{n}}{|\mathbf{r} - \mathbf{r}_i|^3} \tag{7}$$

To solve this integral equation numerically, we can break the boundary into fragments, or elements, and approximate the continuous surface charge distribution by a set of surface charge density values, one for each boundary element (BE). Integrating both sides of Eq. (7) over the boundary element S_k we obtain

$$\iint\limits_{s_k} \sigma\, ds_k - \left(\frac{\varepsilon_{\mathrm{in}}-\varepsilon_{\mathrm{out}}}{2\pi(\varepsilon_{\mathrm{in}}+\varepsilon_{\mathrm{out}})}\right)\iint\limits_{s_k}\iint \frac{\sigma_s(\mathbf{r}_{s_k}-\mathbf{r}_s)\cdot\mathbf{n}_{s_k}}{|\mathbf{r}_{s_k}-\mathbf{r}_s|^3}\, ds\, ds_k = \left(\frac{\varepsilon_{\mathrm{in}}-\varepsilon_{\mathrm{out}}}{2\pi(\varepsilon_{\mathrm{in}}+\varepsilon_{\mathrm{out}})}\right)\frac{1}{\varepsilon_{in}}\sum_i \iint\limits_{s_k} \frac{q_i(\mathbf{r}_{s_k}-\mathbf{r}_i)\cdot\mathbf{n}_{s_k}}{|\mathbf{r}_{s_k}-\mathbf{r}_i|^3}\, ds_k \quad (8)$$

or splitting the integral over the whole surface into a sum of partial integrals and replacing continuous σ with discreet per-element values σ_k

$$\sigma_k A_k - \left(\frac{\varepsilon_{\mathrm{in}}-\varepsilon_{\mathrm{out}}}{2\pi(\varepsilon_{\mathrm{in}}+\varepsilon_{\mathrm{out}})}\right)\sum_j \sigma_j \iint\limits_{s_k}\iint\limits_{s_j} \frac{(\mathbf{r}_{s_k}-\mathbf{r}_{s_j})\cdot\mathbf{n}_{s_k}}{|\mathbf{r}_{s_k}-\mathbf{r}_{s_j}|^3}\, ds_j\, ds_k = \left(\frac{\varepsilon_{\mathrm{in}}-\varepsilon_{\mathrm{out}}}{2\pi(\varepsilon_{\mathrm{in}}+\varepsilon_{\mathrm{out}})}\right)\frac{1}{\varepsilon_{in}}\sum_i q_i \iint\limits_{s_k} \frac{(\mathbf{r}_{s_k}-\mathbf{r}_i)\cdot\mathbf{n}_{s_k}}{|\mathbf{r}_{s_k}-\mathbf{r}_i|^3}\, ds_k \quad (9)$$

where A_k is the area of the kth surface element. This integral equation then turns into a system of linear equations:

$$\mathbf{R}\boldsymbol{\sigma} = \mathbf{e} \quad (10)$$

where σ is the vector of BE charge densities, matrix **R** only depends on the boundary shape and is defined as

$$R_{jk} = \delta_{jk}A_k - \left(\frac{\varepsilon_{\mathrm{in}}-\varepsilon_{\mathrm{out}}}{2\pi(\varepsilon_{\mathrm{in}}-\varepsilon_{\mathrm{out}})}\right)\iint\limits_{s_k}\iint\limits_{s_j} \frac{(\mathbf{r}_{s_k}-\mathbf{r}_{s_j})\cdot\mathbf{n}_{s_k}}{|\mathbf{r}_{s_k}-\mathbf{r}_{s_j}|^3}\, ds_j\, ds_k \quad (11)$$

and vector **e** is defined as

$$e_k = \left(\frac{\varepsilon_{\mathrm{in}}-\varepsilon_{\mathrm{out}}}{2\pi(\varepsilon_{\mathrm{in}}+\varepsilon_{\mathrm{out}})}\right)\frac{1}{\varepsilon_{in}}\sum_i q_i \iint\limits_{s_k} \frac{(\mathbf{r}_{s_k}-\mathbf{r}_i)\cdot\mathbf{n}_{s_k}}{|\mathbf{r}_{s_k}-\mathbf{r}_i|^3}\, ds_k \quad (12)$$

Indexes j and k refer to the BEs and i refers to the point charges.

Implementation

A major problem of the BE method is the necessity of solving the linear equation system with the matrix **R** of the size $N \times N$, where N is the number of boundary elements.

Unsatisfactory performance in terms of speed remains an obstacle for a wider use of the BE method for macromolecular electrostatics calculations. In its simplest form, the method is only practical for the relatively small systems where a few hundred BEs are sufficient. Unfortunately, as the number of surface elements and the matrix size grows, the speed of the matrix inversion can make the BE electrostatic calculations impractical. The size of the matrix is proportional to the squared number of the boundary elements (N^2_{BEV}), which means that one cannot have more than a few thousand surface elements, i.e., 4,000 elements require a matrix of 64 Mbytes in size. Also, the time required for the matrix inversion is proportional to N^3_{BE}, making calculations for large systems very slow.

The second problem can be circumvented if instead of the matrix inversion one uses the iterative solution of the linear system. When done properly, the iterative process usually converges in only a few ($<$10) steps. The time required for one iteration is only proportional to N^2_{BE}, which makes the iterative solution preferable for large matrices. To solve the linear equation system, we use the preconditioned biconjugate gradient method[17] that guarantees a good convergence.[14]

Increasing the size of the boundary elements helps to keep N_{BE} small, but deteriorates the precision. The characteristic dimension of the bumps and pits on the protein surface is close to the radius of an atom, which is about an ångstrom for hydrogen atoms that constitute the majority of the surface atoms. If the surface is triangulated and the triangles are used as boundary elements, these triangles should have sides of less than an angstrom in length to retain atomic details. Bigger triangles may result in exceedingly large errors: some atoms become very close to the triangulated surface and may even get outside of it.

To reduce the number of boundary elements while preserving sufficiently accurate boundary representa-

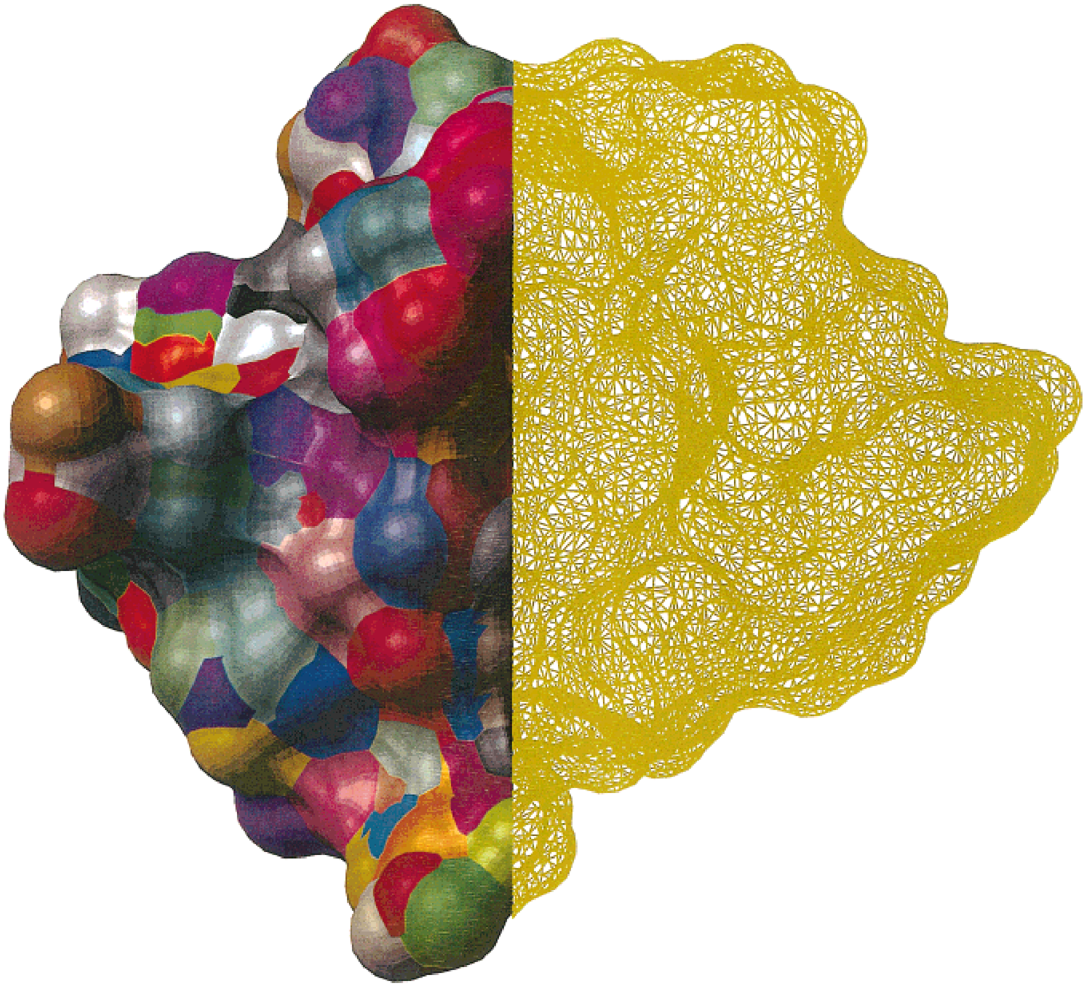


FIGURE 1 The composite boundary elements compared to the basic triangles. Left half of the molecular surface of the polypeptide (shown in solid) has composite boundary elements colored randomly to illustrate the distribution of sizes and shapes. Right half of the molecular surface is displayed as a mesh directly produced by triangulation procedure. Drastic difference in the number of surface elements can be observed.

tion, we define composite boundary elements as associations of all triangles of the surface belonging to one atom. This allows us to have detailed representation of the molecular surface shape with dozens of triangles per surface atom, while the number of boundary elements involved in the linear Eq. (10) is much lower and equals to the number of surface atoms (Fig. 1). The underlying assumption is that the variations of the surface charge density across the atomic surface patch are relatively minor.

Molecular surface was generated by the contour build-up algorithm[18] with subsequent triangulation of the three basic elements (convex spherical patches, toroidal saddles, and concave spherical patches). The algorithm also assigned each triangle to one of the atoms of the molecule. The assignment was used to group the triangles into patches of the surface used as boundary elements. The matrix elements R_{jk} and vector e_k were then calculated by summation of all contributions from the triangular components of patches j and k:

$$R_{j,k} = \sum_{l_j} \sum_{m_k} r_{l_j m_k}, \quad E_k = \sum_{m_k} e_{m_k} \tag{13}$$

where l_j and m_k are indexes of the triangles of the respective patches. To calculate $r_{l,m}$, simple approximation of the integrals in Eqs. (11) and (12) can be obtained by using constant radius-vector $\mathbf{r}_j$ of the point chosen to represent the "center" of the boundary element S_j instead of variable $\mathbf{r}_{sj}$:

$$r_{lm} = \delta_{lm} A_m - \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})} \right) \frac{(\mathbf{r}_l - \mathbf{r}_m) \cdot \mathbf{n}_l}{|\mathbf{r}_l - \mathbf{r}_m|^3} A_l A_m \tag{14}$$

$$e_m = \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})} \right) A_m \frac{1}{\varepsilon_{in}} \sum_i q_i \frac{(\mathbf{r}_m - \mathbf{r}_i) \cdot \mathbf{n}_m}{|\mathbf{r}_m - \mathbf{r}_i|^3} \tag{15}$$

However, this approximation is obviously not valid for the diagonal elements of the matrix, since it is singular for $l = m$. The simplest solution of the singularity problem is to discard the integral for the diagonal elements completely, which would imply that individual surface elements are considered as flat, disregarding their curvature. Some improvement in precision can be achieved if one uses finer tesselation of the surface for the calculation of the diagonal elements.[19] Vorobjev and Scheraga[14] proposed a semianalytic expression for the diagonal elements. The expression, however, included an empiric factor, making the approach somewhat *ad hoc*. Purisma and Nilar[20] suggested that if the surface is composed of interlocking spheres, one can deduce the diagonal elements of the matrix from the off-diagonal ones with the help of a certain normalization condition. We will show now that the normalization condition in fact can be used for arbitrary surface. Indeed, if we consider the sum of the elements of one of the rows of the matrix R, according to Eq. (11):

$$\begin{aligned}
\sum_k R_{jk} &= A_j \\
&\quad - \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})} \right) \sum_k \iint_{s_k} \iint_{s_j} \frac{(\mathbf{r}_{s_k} - \mathbf{r}_{s_j}) \cdot \mathbf{n}_{s_k}}{|\mathbf{r}_{s_k} - \mathbf{r}_{s_j}|^3} ds_j \, ds_k \\
&= A_j - \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})} \right) \iint \iint_{s_j} \frac{(\mathbf{r}_s - \mathbf{r}_{s_j}) \cdot \mathbf{n}_s}{|\mathbf{r}_s - \mathbf{r}_{s_j}|^3} ds_j \, ds \\
&= A_j - \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})} \right) \iint_{s_j} \iint \frac{(\mathbf{r}_s - \mathbf{r}_{s_j}) \cdot \mathbf{n}_s}{|\mathbf{r}_s - \mathbf{r}_{s_j}|^3} ds \, ds_j \\
&= A_j - \left(\frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{2\pi(\varepsilon_{\text{in}} + \varepsilon_{\text{out}})} \right) \iint_{s_j} 2\pi \, ds_j \\
&= \left(1 - \frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{\varepsilon_{\text{in}} + \varepsilon_{\text{out}}} \right) A_j
\end{aligned}$$

or

$$R_{jj} = \left(1 - \frac{\varepsilon_{\text{in}} - \varepsilon_{\text{out}}}{\varepsilon_{\text{in}} + \varepsilon_{\text{out}}} \right) A_j - \sum_{k \neq j} R_{jk}$$

Thus, diagonal elements of the matrix can be deduced from the sums of the off-diagonal elements and the surface area of the corresponding boundary elements. As Purisima and Nilar[20] pointed out, this method of derivation of the diagonal elements assures consistency of the diagonal matrix, which otherwise can become singular or can have negative eigenvalues resulting in large errors.

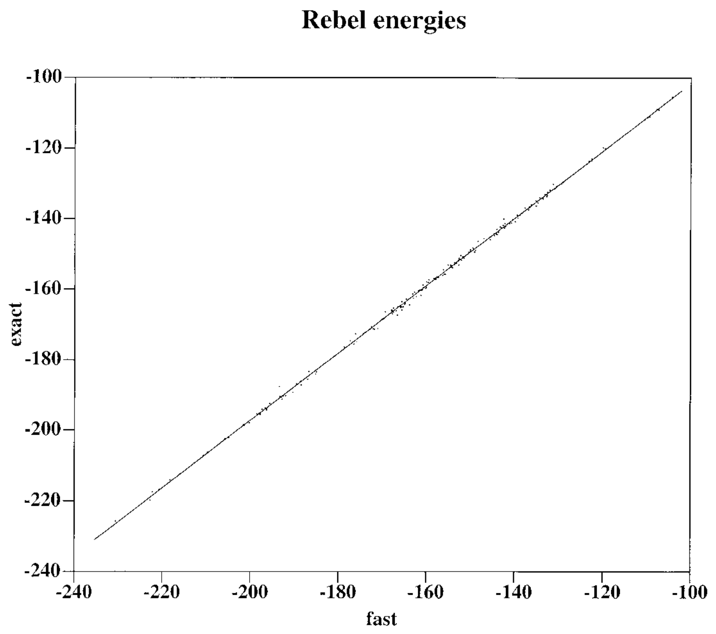
To improve precision further, we also applied surface charge normalization using the relation between the total induced surface charge and total charge of the molecule.[19] To avoid singularity for molecules with zero net charge, calculations are performed twice, only for negative charges and only for positive charges as described by Nicholls and colleagues.[13]

The Setup of the Folding Simulations

The Biased Probability Monte Carlo (BPMC) minimization procedure[8] was used to globally optimize the free energy function in the internal coordinate space. BPMC procedure consists of the following steps performed iteratively:

1. Biased random change of a randomly chosen group of internal variables. Groups changed simultaneously include χ-angles of a single side-chain or φ, ψ pair of the peptide backbone. The probabilities of the new random values of angles are biased according to the database-derived distributions.[8,21] If the φ, ψ change is performed on the residue adjacent to a β-strand, coordinated "β-zipping" move is performed with a certain probability, extending the β-strand.[22]
2. Local gradient energy minimization using pseudo-Newtonian minimizer.
3. Solvation electrostatic energy and surface-based entropic term[8] calculations.
4. Metropolis acceptance criterion: the new conformation is accepted always if it has lower energy, or with the probability of $\exp(-\Delta E / RT_{\text{sim}})$ if it has higher energy then the previous conformation (ΔE is the energy increase and T_{sim} is the simulation temperature, set to 600 K).

Low-energy conformations encountered during the MC search were accumulated in the conformational



Linear regression: Y = 0.956508*X -5.939124 Stand.dev.=0.688771

FIGURE 2 Correlation plot for the solvation electrostatic energies calculated using composite boundary elements and basic triangles.

stack as described in Ref. 23. As a starting point for each MC run, fully randomized conformations were generated with random values in the range of $-180°$ to 180° assigned to all flexible torsion angles in the peptide. To ensure convergence, eight independent MC runs were performed.

The double-energy scheme, which excludes the slow terms from the local minimization, allows us to use efficient gradient minimizer to quickly get near the local minima of the energy function while still including these terms in the global search.[24] During the local minimization step, a modified ECEPP3[22,25] force-field energy with distance-dependent dielectric constant $\varepsilon = 4R$[26,27] was used. Up to 480 energy evaluations were allowed during each local minimization run. At the evaluation step, electrostatic solvation energy replaced the modified Coulomb term. At the same time accessible-surface-based entropic term was added. Dielectric constants of 2 and 78.5 respectively were used for the interior and the exterior of the molecule, respectively. The total number of energy evaluations during the MC run was limited to 12,500,000.

RESULTS AND DISCUSSION

Three factors are crucial for the solvation electrostatic calculations to be included in peptide folding simulations: speed, accuracy, and robustness. Due to the large conformational space, thousands of conformations have to be evaluated to locate the global minimum even with the most efficient search algorithms, requiring fast energy evaluation. The energy difference between the native conformation and the unfolded/misfolded conformations is only a few kcal/mole, thus making errors in energy evaluation beyond 1–2 kcal/mole prohibitive. The large number of energy evaluations and the diversity of conformations during each simulation imposes additional requirements to the accuracy and robustness of the electrostatic calculation.

To address these challenges, we implemented an accurate boundary element solvation electrostatics method, abbreviated as REBEL, with a number of improvements. We proposed the per-atom composite BE definition, which allows to maintain high-resolution boundary representation while keeping the total number of BEs modest. To test the precision of this approximation, we compared the energy values for 200 peptide conformations obtained using the per-atom composite BE and standard flat BE (Figure 2). The rms deviation of the solvation component of the electrostatic energy for the whole molecule calculated by the two methods was as low as 0.68 kcal/mole. The average number of per-atom composite BEs was 273, while the number of standard flat BEs was 2993, more

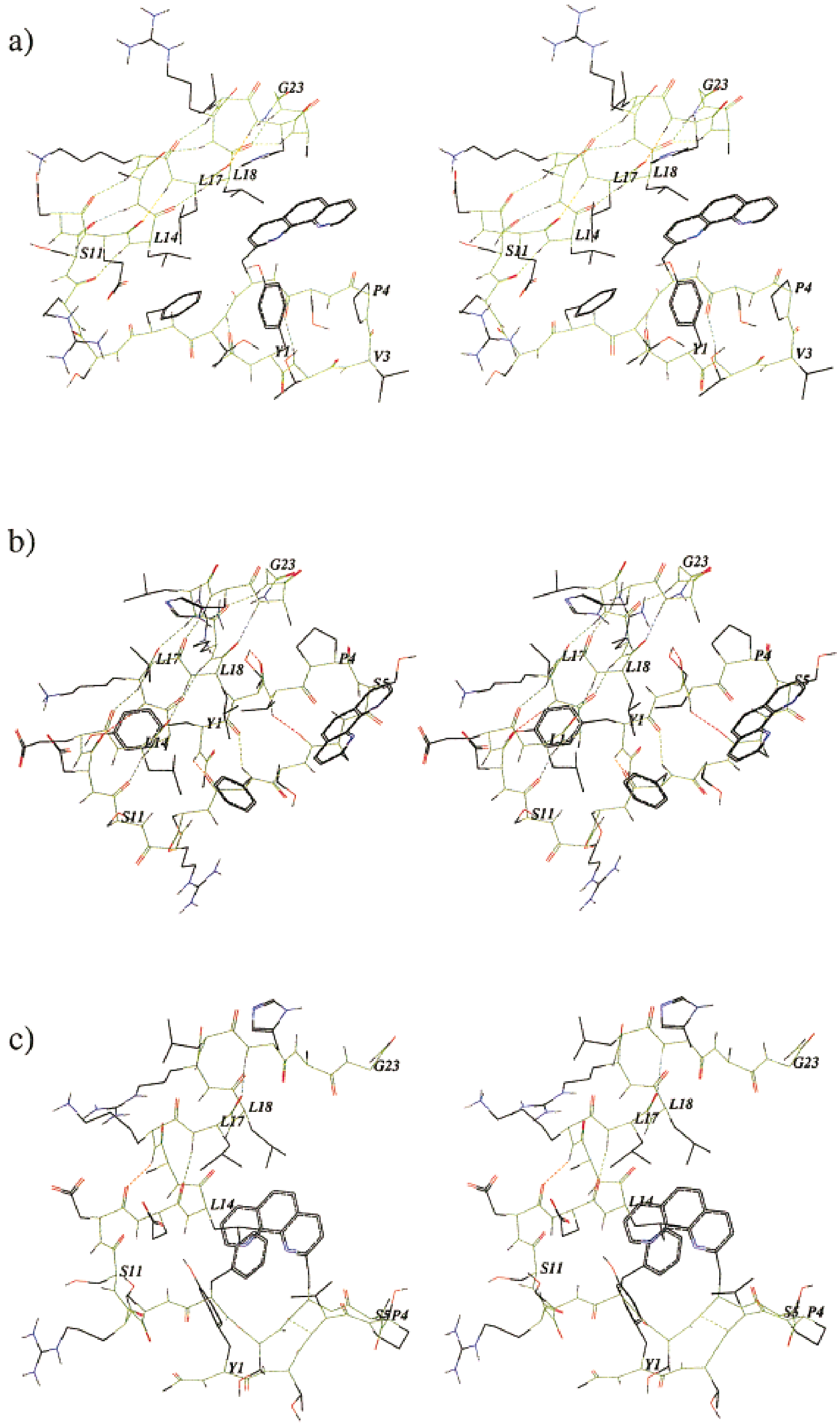


FIGURE 3 Stereo views of simulated and experimental conformations of the 23-residue peptide. Hydrogen bonds between backbone atoms are shown in dotted lines to highlight formation of secondary structure elements. (a) Lowest-energy conformation found during the folding simulations with solvation electrostatics. Contact residues in the α-helix are the same as in NMR structure, however, the β-turn occurs on residues 3 and 4. (b) Low-energy conformation with the same β-turn type as observed in NMR structure, but different helix-sheet interface. (c) NMR structure from Protein Data Bank entry 1HCW.

than a tenfold reduction. Calculation using per-atom composite BE took on average 0.68 s, almost 20 times faster than with standard flat BEs, for which the average time was 13.2 s. Conjugate gradient iterations to solve the BE equation (10) converged after 8–12 steps. We also measured the time spent by the procedure on various stages of the algorithm. For an average-size 129-residue protein lysozyme, the total runtime to calculate the electrostatic energy was 5.6 s. Surface calculation by the contour-buildup algorithm took 1.3 s, matrix preparation 1.9 s, conjugate gradients linear equation solution 1.7 s, and final energy calculations 0.7 s. The time distribution shows that the technique is currently well balanced, with no prominent bottlenecks. All calculations were performed on a single Alpha 677 mHz CPU under Linux operating system.

We generalized the indirect approach of the Purisima and Nilar[20] to the calculation of the diagonal elements of the boundary matrix, proving its applicability to the molecular surface of the arbitrary closed shape. We observed that the direct calculation of the diagonal elements for certain conformations of the polypeptide resulted in an ill-conditioned matrix and, hence, large errors in energy evaluation. Albeit rare, when such situation occurred during the folding simulation, the conformation was assigned an erroneously low energy, misdirecting the conformational search. However, indirect approach substantially improved robustness of the method.

Eight independent peptide folding simulation runs were performed, starting from different random conformations. Each simulation took an average of 40 h on a single Alpha 677 mHz CPU. Figure 3 (a, b) illustrates some low-energy structures found by the procedure. The analysis of the conformational stacks accumulated during the simulations has shown that the lowest-energy conformation, as well as a large fraction of other conformations within 5 kcal/mole from the lowest energy, exhibit a well-defined $\beta\beta\alpha$-fold. The helical segment starts at S11, in agreement with the experimental structure[16] (Figure 3c). In the lowest-energy conformation, the β-sheet packs correctly against the hydrophobic face on the α-helix, making contacts with L14, L17, and L18, however the type of the β-turn formed differs from the one predicted in the consensus NMR structure. In the latter, β-turn occurs on residues P4 and S5, while in the lowest energy conformation it is shifted to V3 and P4 (Figure 3a), resulting in overall backbone rms deviation (RMSD) 3.4 Å. One possible explanation for this discrepancy is that our energy function overestimates the free energy gain from the burial of the large Fen side chain, which is almost completely buried in the predicted conformation and largely exposed in the NMR structure. Several other low-energy $\beta\beta\alpha$ conformations have the reported type of the β-turn (backbone RMSD for the first 8 residues 1.2 Å), but the helix–sheet interface residues differ significantly from the experimental structure. The helix is rotated around its axis by about 90°, leaving L17 completely out of the interface.

We conclude that our folding protocol does find most of the features of the native structure correctly. The imperfect agreement with the experimental structure may reflect shortcomings of the force field used as well as low resolution of the experimental data and limitations of a single conformation representations of a conformational ensemble.

To illustrate the importance of the solvation term, we also performed "control" simulations with the electrostatic term replaced by a distance-dependent dielectric constant calculation. Low-energy conformations found in the control runs had neither β-sheet nor α-helix secondary structure elements formed, indicating that accurate solvation contribution is essential in the energetics of peptide folding.

CONCLUSIONS

An efficient implementation of the boundary element method for solvation electrostatics calculations is proposed. A combination of three improvements, namely the use of large composite boundary elements, the sum rule for diagonal element calculation and the surface charge normalization algorithm, made the method sufficiently fast (less than 1 s per calculation), accurate (error RMS 0.7 kcal/mole as compared to much more detailed but slower implementation) and robust (less than one failure per 200,000 energy calculations) to be used directly in the folding simulations of a 23-residue peptide for the first time. The BPMC procedure with β-zipping step and exact electrostatic solvation term successfully predicted formation of the two basic elements of the protein structure, β-sheet as well as α-helix and favored their correct packing. The electrostatic solvation term is shown to be critical for correct folding.

We would like to thank Compaq for providing Molsoft with the Alpha workstation, which was used for the calculation and analysis.

REFERENCES

1. Simmerling, C. L.; Elber, R. Proc Natl Acad Sci USA 1995, 92, 3190–3193.

2. Mohanty, D.; Elber, R.; Thirumalai, D.; Beglov, D.; Roux, B. J Mol Biol 1997, 272(3), 423–442.
3. Srinivasan, J.; Miller, J.; Kollman, P. A.; Case, D. A. J Biomol Struct Dyn 1998, 16(3), 671–682.
4. Cheatham, T. E., III; Kollman, P. A. Ann Rev Phys Chem 2000, 5, 435–471.
5. Lee, M. R.; Duan, Y.; Kollman, P. A. Proteins 2000, 39(4), 309–316.
6. Friedman, H. L. Mol Phys 1975, 29, 1533–1543.
7. Schaefer, M.; Froemmel, C. J Mol Biol 1990, 216, 1045–1066.
8. Abagyan, R. A.; Totrov, M. M. J Mol Biol 1994, 235, 983–1002.
9. Bashford, D.; Case, D. A. Ann Rev Phys Chem 2000, 51, 129–152.
10. Nicholls, A.; Honig, B. J Comput Chem 1991, 12, 435–445.
11. Zauhar, R. J.; Morgan, R. S. J Mol Biol 1985, 186, 815–820.
12. Juffer, A. H.; Botta, E. F. F.; van Keulen, B. A. M.; van der Ploeg, A.; Berendsen, H. J. C. J Comput Phys 1991, 97, 144–171.
13. Bharadwaj, A.; Windemuth, A.; Sridharan, S.; Honig, B.; Nicholls, A. J Comp Chem 1995, 16, 898–910.
14. Vorobjev, N. Y.; Scheraga, H. A. J Comp Chem 1997, 18(4), 569–583.
15. Ripoll, D. R.; Vila, J. A.; Villegas, M. E.; Scheraga, H. A. J Mol Biol 1999, 292(2), 431–440.
16. Struthers, M. D.; Cheng, R. P.; Imperiali, B. Science 1996, 271, 342–345.
17. Press, W. H.; et al. 1992. Numerical Recipes in C. Cambridge University Press: New York.
18. Totrov, M. M.; Abagyan, R. A. J Struct Biol 1996, 116, 138–143.
19. Rashin, A.; Namboodiri, K. J Phys Chem 1987, 91, 6003–6012.
20. Purisima, E. O.; Nilar, S. H. J Comp Chem 1995, 16, 864–870.
21. Zhou, Y.; Abagyan, R. In Rigidity Theory and Application; Thorpe, M. E., and Duxbury, P. M., eds.; 1999, 345–356.
22. Abagyan, R.; Totrov, M. J Comp Phys 1999, 151, 402–421.
23. Abagyan, R. A.; Argos, P. J Mol Biol 1992, 225, 519–532.
24. Abagyan, R. A.; Totrov, M. M.; Kuznetsov, D. A. J Comp Chem 1994, 15, 488–506.
25. Nemethy, G.; Gibson, K. D.; Palmer, K. A.; Yoon, C. N.; Paterlini, G.; Zagari, A.; Rumsey, S.; Scheraga, H. A. J Phys Chem 1992, 96, 6472–6484.
26. McCammon, J. A.; Wolynes, P. G.; Karplus, M. Biochemistry 1979, 18, 927–942.
27. Pickersgill, R. W. Protein Eng 1988, 2, 247–248.
28. Lee, B.; Richards, F. M. J Mol Biol 1971, 55, 379–400.
29. Shrake, A.; Rupley, J. A. J Mol Biol 1973, 79, 351–371.
30. Richards, F. M. Ann Rev Biophys Bioeng 1977, 6, 151–176.
31. Lorensen, W.; Cline, H. Computer Graphics 1987, 21, 163–169.
32. Connolly, M. L. J Appl Cryst 1983, 16, 548–558.