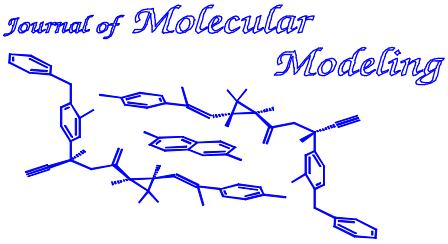
J. Mol. Model. **1995**, *1*, 1 – 10



Folding of Globular Proteins by Energy Minimization and Monte Carlo Simulations with Hydrophobic Surface Area Potentials

Christian Mumenthaler and Werner Braun*

Institut für Molekularbiologie und Biophysik, Eidgenössische Technische Hochschule, Hönggerberg, CH - 8093 Zürich, Switzerland (braun@mol.biol.ethz.ch)

Received: 1 December 1994 / Accepted: 4 January 1995

Abstract

We describe an efficient method for the analytical calculation of the solvent accessible surface areas and their gradients in proteins on serial and parallel computers. We applied energy minimizations and Monte Carlo simulations to the small three-helix bundle protein E*r*-10. The force field consisted of the ECEPP/2 energies and a term describing protein-solvent interaction through the solvent accessible surface area. We show that the NMR structure is stable when refined with this force field, but large structural changes are observed in energy minimization in vacuo. When we started from random tertiary structures with preformed helices and maintained the helical segments by dihedral angle constraints, the final structures with the lowest energies resembled the native fold. The root-mean-square deviations for the backbone atoms of the three helices compared to the experimentally determined structure were 3 Å to 4 Å.

Keywords: Protein folding, accesible surface areas, Monte Carlo simulations, FANTOM, parallel computers, three-helix bundle.

Introduction

Anfinson's hyphothesis that the native fold of a protein corresponds to a state of minimal free energy [1] lead to great efforts to establish accurate and reliable empirical force field parameters for proteins [2]. Although several force field parameters have been proposed in the last two decades [3,4,5], we still cannot compute the native fold of a protein on the basis of energetic considerations. Apart from the computational difficulty of locating the global minimum of a function with myriads of local minima, there is the problem of modelling the protein-solvent interaction. This interaction contributes significantly to the stability of the native fold of a protein[6]. Examples of deliberately misfolded proteins clearly demonstrate that only the inclusion of protein- solvent interactions makes it possible to recognize the correct fold among other alternative folds [7].

Protein-solvent interactions have been modeled through the accessible surface areas of individual atoms[8,9,10] which can be calculated analytically [11,12] or numerically [13]. For efficient energy minimization and molecular dynamics calculation the gradient of the solvent accessible surface areas with respect to Cartesian coordinates or torsion angles has to be calculated analytically [14,15,16]. Detailed mathematical descriptions for the correct calculation of the gradient have been published recently [16].

In this paper, we show that this calculation can be further simplified by deriving new and computationally more efficient equations. These equations were integrated into the energy minimization and Monte Carlo simulation package FANTOM [17,18]. The solvent accessible area and its gradient is calculated twice as fast, the solvent accessible surface alone

* *To whom correspondence should be addressed*

three times faster compared to our previous routine [16]. The procedure can also be ported efficiently to parallel computers which get more and more popular in the field of scientific computing [19].

The influence of a solvation term in energy minimizations and Monte Carlo simulations has been described for a few proteins: bovine pancreatic trypsin inhibitor (BPTI) [20,21], α-amylase inhibitor (Tendamistat) [21], and avian pancreatic polypeptide [22]. We have shown [21] that atomic solvation parameters based on a simple polar/nonpolar classification of atoms drive perturbed NMR structures of BPTI and Tendamistat back to the native structures. In contrast, calculations with previously published parameters [9,15] did not correct the perturbations.

We have now applied our parameters in a folding study of the three-helix bundle E*r*-10. Similar to other studies [23,24] our primary interest is the correct packing and not the formation of helical regions. Therefore the helical regions determined by NMR were specified by dihedral angle constraints in all calculations. Starting from random tertiary structures, the three structures with lowest energies obtained by Monte Carlo simulations including hydrophobic surface terms have the correct fold.

Calculation of the solvent accessible surface area and its gradient

In the continuum approximation the protein-solvent interaction E_{hyd} is computed by

$$E_{hyd} = \sum_{i=1,atoms} \sigma_i A_i \qquad (1)$$

where A_i is the solvent accessible surface area (Fig. 1) of atom i and σ_i is a "solvation-parameter" [8,9,10] depending on the atom type.

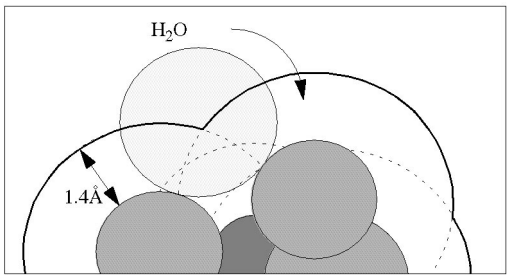


Fig. 1: *Definition of the solvent accessible surface. A sphere with a radius of 1.4 Å representing a water molecule is rolled over the van der Waals surface of the protein. Atoms located in cavities (dark grey) are not touched and therefore considered as buried. The procedure is equivalent to calculating the protein surface where 1.4 Å have been added to the van der Waals radii of all atoms.*

Like other analytical calculations of the solvent accessible surface area and its gradient, our method is based on the global Gauss-Bonnet theorem for the case of intersecting spheres [11,12]:

$$A_i = r^2 \left[2\pi\chi + \sum_{\lambda=1,p} \Omega_{\lambda,\lambda+1} + \sum_{\lambda=1,p} \cos\Theta \cdot \Phi \right] \qquad (2)$$

The solvent accessible surface of atom i is enclosed by p intersecting arcs λ with other spheres. The variables Ω, cosΘ and Φ are defined as in Fig.2 and Fig.3.

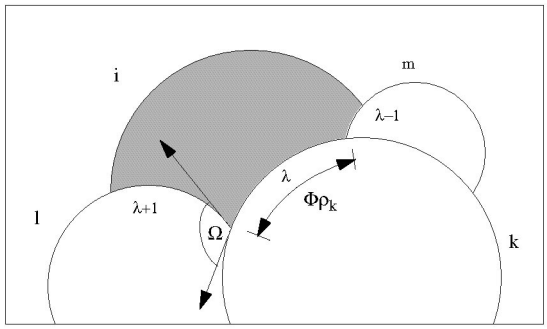


Figure 2: *The solvent accessible surface of atom i is enclosed by a certain number of arcs λ which are parts of the different intersection circles with other atoms and meet in the common intersection points. Ω is the angle between the two tangential vectors in these points, Φ is the angle defining the arc length and ρ_k is the radius of the intersection circle.*

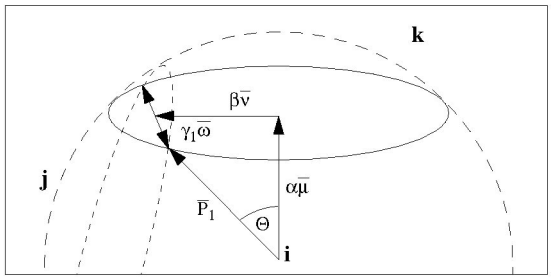


Figure 3: *The vector $\vec{P}_1$ from the origin (center of atom i) to the common intersection point of the atoms i, k and j can be decomposed into three orthogonal vectors $\alpha\vec{\mu}$, $\beta\vec{\nu}$ and $\gamma\vec{\omega}$.*

The analytical calculation of A_i and its derivatives with respect to the coordinates of the intersecting spheres is different from previous work. Compared to Connolly's approach [11], we chose a different representation of the intersection points, and we will give explicit formula for the derivatives. In contrast to Richmond's approach [12] we chose Cartesian coordinates rather than polar coordinates in multiple rotated frames.

We set the sphere i, which is cut by two other spheres, k and j, in the origin of the coordinate system. The center of any other sphere k is then determined by the vector $\vec{x}_k = \left(x^1, x^2, x^3\right)$ with $|\vec{x}_k| = d_k$. The sphere radius is r_k. The three spheres intersect in the two points $\vec{P}_1$ and $\vec{P}_2$ which can be decomposed into three orthogonal vectors:

$$\vec{P}_{1,2} = \alpha\vec{\mu} + \beta\vec{\nu} + \gamma_{1,2}\vec{\omega} \tag{3}$$

α, β, γ_1 and γ_2 are scalars and $\vec{\mu}, \vec{\nu}, \vec{\omega}$ are orthonormal vectors. The vector $\vec{\mu}$ points from the center of sphere i to sphere k and $\vec{\nu}$ is orthogonal to $\vec{\mu}$, pointing to sphere j (Fig. 3). Note that $\gamma 1 = -\gamma 2$ and therefore $\vec{P}_2 = \vec{P}_1 - 2\gamma_1\vec{\omega}$.

For convenience we introduce g_k, the distance from the center of sphere i to the center of the intersection circle with sphere k and the angle φ between the vectors $\vec{x}_j$ and $\vec{x}_k$:

$$g_k = \frac{d_k^2 + r_i^2 - r_k^2}{2d_k} \tag{4}$$

$$\cos\varphi = \frac{\vec{x}_j \vec{x}_k}{d_j d_k} = \frac{\vec{x}_j}{d_j}\vec{\mu} \tag{5}$$

The scalars α, β, γ can be obtained by solving the equations describing the intersection points:

$$\left|\begin{array}{rcl} \left(\vec{x}_j - \vec{P}\right)^2 & = & r_j^2 \\ \left(\vec{x}_k - \vec{P}\right)^2 & = & r_k^2 \\ \vec{P}^2 & = & r_i^2 \end{array}\right| \tag{6}$$

Using the relations $\vec{x}_j^2 = d_j^2$, $\vec{x}_k^2 = d_k^2$, $\vec{x}_k\vec{\nu} = 0$, $\vec{x}_k\vec{\omega} = 0$ and $\vec{x}_j\vec{\omega} = 0$ leads to

$$\begin{aligned} \alpha &= g_k \\ \beta &= \frac{1}{\sin\varphi}\left(g_j - g_k\cos\varphi\right) \\ \gamma_{1,2} &= \pm\sqrt{r_i^2 - \alpha^2 - \beta^2} \end{aligned} \tag{7}$$

The vectors $\vec{\mu}$ and $\vec{\omega}$ are readily obtained and $\vec{\nu}$ has to be constructed as being perpendicular to $\vec{\mu}$

$$\begin{aligned} \vec{\mu} &= \frac{\vec{x}_k}{d_k} \\ \vec{\nu} &= \frac{1}{\sin\varphi}\left(\frac{\vec{x}_j}{d_j} - \cos\varphi\vec{\mu}\right) \end{aligned} \tag{8}$$

$$\vec{\omega} = \vec{\mu} \wedge \vec{\nu}$$

The symbol "$\wedge$" denotes the cross product of two vectors.

For the calculation of the gradient, we need the derivatives of $\vec{P}_1$ and $\vec{P}_2$ with respect to the coordinates of spheres k and j. As α, β, $\vec{\mu}$ and $\vec{\nu}$ are not symmetric with respect to the spheres k and j, their derivatives with respect to the coordinates of k and j are different. One could also use the same formulas as above by exchanging k and j. Then, however, α, β, $\vec{\mu}$ and $\vec{\nu}$ would have to be recalculated.

First, we calculate the derivative of the intersection points with respect to the coordinates of sphere k. The derivatives of all the scalars α, β and γ and all the vectors $\vec{\mu}$, $\vec{\nu}$ and $\vec{\omega}$ in (3) have to be considered. We will use the short hand notation $\partial = \frac{\partial}{\partial x_k^m}$ to keep the equations concise.

Because $\partial\vec{x}_j = 0$ and $\partial d_j = 0$ the derivatives of d_k and $\cos\varphi$ are:

$$\partial d_k = \frac{x_k^m}{d_k} \tag{9}$$

$$\partial\cos\varphi = \partial\left(\vec{\mu}\cdot\frac{\vec{x}_j}{d_j}\right) = \frac{\vec{x}_j}{d_j}\partial\vec{\mu} \tag{10}$$

Because $\varphi < \pi$ we have $\sin\varphi = \sqrt{1-(\cos\varphi)^2}$; thus the derivative of $\sin\varphi$ can be related to the derivative of $\cos\varphi$:

$$\partial\sin\varphi = -\left(\frac{\cos\varphi}{\sin\varphi}\right)\partial\cos\varphi \tag{11}$$

With equations (9), (10) and (11), the derivatives of α, β and γ are easily obtained:

$$\partial\alpha = \partial g_k = \partial\left(\frac{d_k^2 + r_i^2 - r_k^2}{2d_k}\right) = \left(\frac{d_k - g_k}{d_k}\right)\partial d_k \tag{12}$$

$$\partial\beta = \frac{1}{\sin\varphi}\left[-\partial g_k\cos\varphi - g_k\partial\cos\varphi - \beta\partial\sin\varphi\right] \tag{13}$$

$$\partial\gamma_{1,2} = \partial\left(\pm\sqrt{r_i^2 - \alpha^2 - \beta^2}\right) = -\frac{1}{\gamma_{1,2}}\left(\alpha\partial\alpha + \beta\partial\beta\right) \tag{14}$$

In eq. (13) we introduced β as previously calculated by equation (7).

For the vectors, we get

$$\partial\vec{\mu} = \frac{1}{d_k^3}\vec{\mu} + \frac{1}{d_k}\begin{bmatrix}\delta_{1,m}\\ \delta_{2,m}\\ \delta_{3,m}\end{bmatrix} \tag{15}$$

where the m in the Kronecker-symbol indicates the component of x_k m.

The derivative of $\vec{\nu}$ is calculated analogously to $\partial\beta$:

$$\partial\vec{\nu} = \frac{1}{\sin\varphi}\left[-\vec{\mu}\partial\cos\varphi - \partial\vec{\mu}\cos\varphi - \vec{\nu}\partial\sin\varphi\right] \tag{16}$$

Finally, $\partial\vec{\omega}$ is obtained by:

$$\partial\vec{\omega} = \partial\vec{\mu}\wedge\vec{\nu} + \vec{\mu}\wedge\partial\vec{\nu} \tag{17}$$

For the calculation of the gradients with respect to the coordinates of sphere j we will give the final equations by using the notation $\partial_j = \frac{\partial}{x_j^m}$:

$$\partial_j\alpha = 0 \tag{18}$$

$$\partial_j\beta = \frac{1}{\sin\varphi}\left[\partial_j g_j - g_k\partial_j\cos\varphi - \beta\partial_j\sin\varphi\right] \tag{19}$$

$$\partial_j\gamma = -\frac{1}{\gamma}\left(\beta\partial_j\beta\right) \tag{20}$$

The derivative of $\vec{\mu}$ vanishes and for the vectors $\vec{\nu}$ and $\vec{\omega}$ we find:

$$\partial_j\vec{\nu} = \frac{1}{\sin\varphi}\left[\partial_j\left(\frac{\vec{x}_j}{d_j}\right) - \vec{\mu}\partial_j\cos\varphi - \vec{\nu}\partial_j\sin\varphi\right] \tag{21}$$

$$\partial_j\vec{\omega} = \vec{\mu}\wedge\partial_j\vec{\nu} \tag{22}$$

With the equations for the intersection points and their derivatives, we can now calculate the values cosΘ, Φ and Ω needed by the Gauss-Bonnet formula. As illustrated in Fig. 4, Φ and Ω can be easily obtained through the tangential vectors $\vec{n}_{ijk}^{(p)}$ in the intersection points.

The cosine of the polar angle Θ can be calculated directly as the distance from the center of sphere i to the center of the intersection circle k is α and $\left|\vec{P}_1\right| = r_i$:

$$\cos\Theta = \frac{\alpha}{r_i} \tag{23}$$

For the tangential vectors $\vec{n}_{ijk}^{(p)}$, we will use the following notation: The three indices ijk label the three spheres which intersect in the considered point. The first two indices, i and j, give the intersection circle to which the vector is tangential. The number of the intersection point p (1 or 2) is written in superscript. The intersection points can be classified as "entry" or "exit" points. These are the points where one enters or leaves the buried arc when moving on the oriented intersection circle. In Fig. 4, $\vec{P}_1$ is an exit point of the intersection circle k.

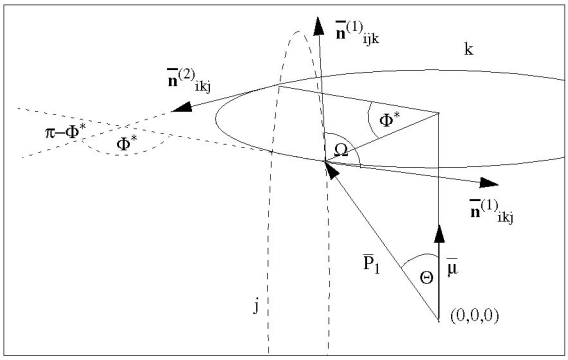


Figure 4: *Atom i lies in the origin and is cut by the atoms k and j. Positively oriented tangential vectors are drawn in the intersection points* $\vec{P}_1$ *and* $\vec{P}_2$*. Both,* Φ *and* Ω *can be calculated as scalar products of these vectors. In this drawing,* Φ* *is the complementary angle of* Φ *(*Φ=2π-Φ**).*

As the angle of the accessible part of the intersection circle Φ can be larger than π, the scalar product of the tangential vectors $\vec{n}_{ijk}^{(1)}$ and $\vec{n}_{ijk}^{(2)}$ can be either Φ or the complementary angle Φ*=2π-Φ. To distinguish between the two angles, we have to introduce a new vector $\vec{h}$:

$$\vec{h} = \vec{n}_{ikj}^{(1)}\wedge\vec{n}_{ikj}^{(2)} \tag{24}$$

If Φ<π this vector will point in the same direction as $\vec{x}_k$. The scalar product of $\vec{h}$ with $\vec{x}_k$ will therefore determine the calculation of Φ:

$$s = \vec{h}\cdot\vec{x}_k$$

$$\Phi = \begin{cases}\operatorname{acos}\left(\vec{n}_{ikj}^{(1)}\cdot\vec{n}_{ikj}^{(2)}\right) & s \geq 0\\ 2\pi - \operatorname{acos}\left(\vec{n}_{ikj}^{(1)}\cdot\vec{n}_{ikj}^{(2)}\right) & s < 0\end{cases} \tag{25}$$

Ω can not be larger than π, thus

$$\Omega = \operatorname{acos}\left(\vec{n}_{ikj}^{(1)}\cdot\vec{n}_{ijk}^{(1)}\right) \tag{26}$$

The tangential vectors themselves can be calculated with the intersection points $\vec{P}_1$ and $\vec{P}_2$ which we have previously calculated:

$$\vec{n}_{ikj}^{(1,2)} = \frac{\vec{\mu}\wedge\vec{P}_{1,2}}{r_i\sin\Theta} \tag{27}$$

$$\vec{n}_{ijk}^{(1,2)} = \frac{\vec{x}_j \wedge \vec{P}_{1,2}}{d_j r_i \sin\Theta} \tag{28}$$

For the calculation of the gradient of the solvation energy (1), we need the derivative of every solvent accessible surface A_i with respect to all atom coordinates. The matrix $\frac{\partial A_i}{\partial x_k}$ is, however, sparse, as only those derivatives are different from zero where the sphere k does cut the sphere i. From the Gauss-Bonnet theorem (2) we have

(29)

$$\frac{\partial A_i}{\partial x_k} = r^2 \left[\sum_{\lambda=1,p} \frac{\partial \Omega_{\lambda,\lambda+1}}{\partial x_k} + \sum_{\lambda=1,p} \frac{\partial \cos\Theta}{\partial x_k} \cdot \Phi + \sum_{\lambda=1,p} \cos\Theta \frac{\partial \Theta}{\partial x_k} \right]$$

If the sphere k has only one accessible arc λ on sphere i, the first sum contains two terms (angles at the start and end of the arc), the second sum only one term (cosΘ depends only on k) and the last sum three terms (the Φ of the arcs λ-1 and λ+1 do also depend on the location of sphere k).

Again, we will first calculate the derivatives with respect to k using the notation $\partial = \frac{\partial}{\partial x_k^m}$

For the main terms in (29), we get:

$$\partial\Omega = \frac{1}{\sin\Omega}\left(\partial\vec{n}_{ikj}^{(1)} \cdot \vec{n}_{ijk}^{(1)} + \vec{n}_{ikj}^{(1)} \cdot \partial\vec{n}_{ijk}^{(1)}\right) \tag{30}$$

$$\partial\cos\Theta = \frac{\partial\alpha}{r_i} \tag{31}$$

$$\begin{aligned}\partial\Phi &= \partial\,\mathrm{acos}\left(\vec{n}_{ikj}^{(1)} \cdot \vec{n}_{ikj}^{(2)}\right) \\ &= \frac{1}{\sin\Phi}\left[\partial\vec{n}_{ikj}^{(1)} \cdot \vec{n}_{ikj}^{(2)} + \vec{n}_{ikj}^{(1)} \cdot \partial\vec{n}_{ikj}^{(2)}\right]\end{aligned} \tag{32}$$

Note that the formula for Φ is given for the case that s>0 (see above). For s<0, the formula has to be multiplied by -1. The derivatives of the tangential vectors given in (27) and (28) can again be expressed by previously calculated terms.

Implementation in FANTOM

We have integrated the calculation of surface areas and their gradients into the program FANTOM version 3.5. The new Fortran routine PARAREA replaces the previous routine SAREA [16] and performs all calculations using the atom coordinates and radii as input [25].

In a first step, a list of intersecting atoms k are generated for every atom i. Atoms whose intersection circles with i are entirely contained in another one are removed from the list at this early stage. The number of atoms in this intersection list is only dependant on the average packing density of the protein and not on the overall size of the protein. Thus, the major part of the CPU time needed for the calculations increases linearly with the size of the protein.

In a second step, the existing intersection points of the atoms i, k and j are calculated with equations (7) and (8). As every intersection point is either an “entry” or an “exit” point on the oriented intersection circle k, it is easy to determine which points (if any) delimit an accessible arc on this circle and which points are buried.

Finally, we can calculate all values needed for the Gauss-Bonnet formula in equations (2) and (29) for each of these arcs λ. An arc λ of the intersection circle is delimited by two points $\vec{P}_{ikj}$ and $\vec{P}_{ikm}$ where j and m are typically two different atoms. These two intersection points and their derivatives can be calculated by the same equations as j and m are mathematically analogous.

The calculated accessible areas and gradients have been extensively tested with values obtained with the previous FANTOM routine SAREA [16]. We have also compared the analytical to the numerical gradient for different structures. All values agreed within the accuracy of the numerical gradient (Table 1).

Table 1: *CPU times and accuracy of PARAREA in the calculation of the solvent accessible surface and the gradient for Tendamistat* [*a*]

Computer	**CPU [sec] Area**	**CPU [sec] Area & Gradient**	**D(E_{hyd})[b]**
Sun Sparc/2	11	14	$1.32*10^{-6}$
Cray Y-MP	0.78	1.13	$1.32*10^{-6}$
Paragon [c]	0.46	0.52	1.32*10-6

[*a*] *All 558 heavy atoms of the protein Tendamistat (74 residues) were included.*
[*b*] *Relative difference of the numerical gradient to the analytical gradient. We used a displacement of 10^{-4} Å for the calculation of the numerical gradient.*
[*c*] *30 slave processors were used for this calculation.*

Parallel computers

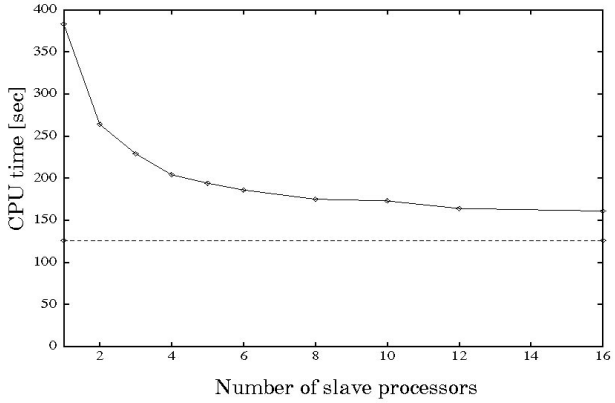
PARAREA was also ported to an Intel Paragon distributed memory parallel computer. The calculations of the individual solvent accessible surfaces and their gradients are intrinsically

parallel as every atom can be treated separately provided all coordinates and radii of the other atoms are known. The available processors are divided into n-1 "slave" processors which perform the computations and one "master" processor which sends the initial data to the slaves and calculates the solvation energy and it's gradient as soon as the results are sent back by the slaves. Therefore, in a protein with M atoms, every slave processor calculates the solvent accessible surface of M/(n-1) atoms.

As some parts of the protein might be buried and require much less CPU time, assigning equal sequential fragments of the protein to every slave processor would yield a bad load balance. We achieved an almost optimal load balance by assigning every (n-1)th atom to the same slave processor.

The CPU times needed for 50 minimization steps of the protein E*r*-10 (38 residues) with and without solvation energy term (Figure 5) show that the solvation energy does not dominate the calculations any more as soon as a few processors are used. With one processor the calculation of the solvation energy needs 67% of the total CPU time. With 20 slave processors it drops to 22%. These fractions decrease with increasing protein size. They are 59% and 10% for Tendamistat (74 residues).

The fastest algorithm for the calculation of the accessible surface area, MSEED [14], has been reported to have similar properties. The CPU time needed for the calculation of the solvation energy was 59% of the total CPU time in a minimization of the 5 residue peptide Met-enkephalin. However, MSEED's search for accessible intersection points on the protein surface is a recursive algorithm which makes an efficient parallelization very difficult. Furthermore, it does not take into account internal cavities or intersection circles



which are not cut by a third sphere. Both points could lead to problems during the minimization of unfolded structures.

Figure 5: *CPU times on the Intel Paragon computer needed for 50 energy minimization steps including the solvation term of an unfolded Er-10 protein. The dashed line refers to a corresponding energy minimization in vacuo.*

Folding studies with E*r*-10

Computational details. For our folding studies, we have used the pheromone E*r*-10 from the ciliated protozoan *Euplotes raikovi*. The tertiary structure of this protein was solved by NMR spectroscopy [26]. It has 38 amino acid residues and is folded in a three-helix bundle which is stabilized by three Cys-Cys disulfide bridges between residues 3-19, 10-37 and 15-27. We used model 1 of the atomic coordinate file 1ERP in the Brookhaven Protein Data Bank [27] as NMR reference structure. All root-mean-square deviations (rmsd) of the calculated structures are given for the backbone atoms in the helical regions with respect to this reference structure.

We did not restrain the disulfide bridges in our study, but the helices were already formed in the initial unfolded structures. We assigned the segments 2-8, 12-18, and 24-32 as helices based on the NMR work [26] with the exception of residues 19 and 33 which have Ψ angles largely deviating from the typical α-helical value. The backbone dihedral

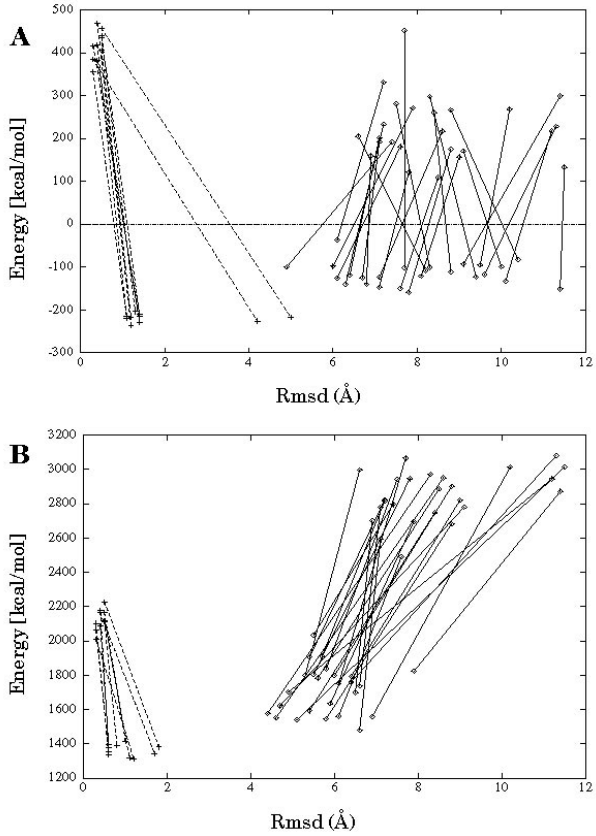


Figure 6: *Energy minimizations of NMR structures (dashed lines) and unfolded structures (full lines) with the ECEPP/2 force field alone (A) and the ECEPP/2 force field including a highly weighted hydrophobic energy term (B). Every line connects the initial structure (high energy) to the final structure after 500 minimization steps (lower energy). The rmsd values have been calculated with respect to the three helices of the NMR reference structure.*

angles of the residues in the helices were restrained to $-72^{o}<\Phi<-42^{o}$ and $-62^{o}<\Psi<-32^{o}$ in all calculations.

An ensemble of 25 initial, unfolded structures with preformed helices were generated by the distance geometry program DIANA [28]. Restrained energy minimizations with the helical dihedral angle constraints were then performed with the program package FANTOM [17,18], using the APOLAR solvation parameters defined in our previous work [21]. The solvation parameters σ_i were set to 1 kcal $mol^{-1}Å^{-2}$ for carbon and sulphur atoms and to zero for all other atoms.

The sulphur atoms of cysteines were defined as hydrophobic to favour the burying of the cysteines in the protein core. The van der Waals radii were taken from Table 2 of the work of Shrake and Rupley [13].

Energy minimizations consisted of 500 iteration steps of the conjugate gradient method. We have used a 8 Å cutoff value for the nonbonded interaction list which was updated every 10 minimization steps. The parameters for the minimization σ, ρ and τ were set to 0.4, 0.4 and 0.1, respectively. To avoid singularities in the ECEPP/2 force field we used a smoothed Lennard-Jones potential for nonbonded distances smaller than 2.0 Å. The dielectric constant was set to be proportional to interatomic distances.

We also performed the same calculations starting from 10 unrefined NMR structures which had backbone rmsd values of less than 0.5 Å compared to the NMR reference structure. All energy minimizations were repeated with the ECEPP/2 potential in vacuo.

As a second method to locate low energy conformations we applied the combination of Monte Carlo simulation and energy minimization of Li and Scheraga [29] modified with an adaptive temperature schedule [18]. We chose 10 structures of the 25 initial DIANA structures which had rmsd values ranging from 6 Å to 11 Å. 160 Monte Carlo steps using the Metropolis criterion were performed with 50 energy minimizations each. The energy function was the same as described above. Only the backbone angles Φ and Ψ of the 7 loop residues located between the helices were selected to be variable in the Monte Carlo step and only one angle was changed every step. During the first 80 steps, every angle could change within a range of 180^{o} to allow large structural modifications. During the second 80 steps this range was lowered to 30^{o}. The adaptive temperature schedule [18] during the Monte Carlo simulations was as follows: The initial temperature of 300 K was lowered to 5 K every time a conformation with lower energy was found or raised by 500 K if no conformation with lower energy was found within the last 10 Monte Carlo steps.

Finally, the Monte Carlo structures were minimized to a local minimum with reduced protein-solvent parameters of 25 cal mol^{-1} $Å^{-2}$ for the carbon and sulphur atoms. These values correspond to standard estimates of the hydrophobic contribution to the protein solvation energy based on studies with hydrocarbons [30]. The structures were assumed to be in a local minimum, if their energies did not decrease more than 0.1% or 10^{-4} kcal mol^{-1} in the last 50 minimization steps. Depending on the structure, 600 to 4000 minimization steps were necessary. Three of the energy refined NMR structures were minimized with the same potential to obtain reference value.

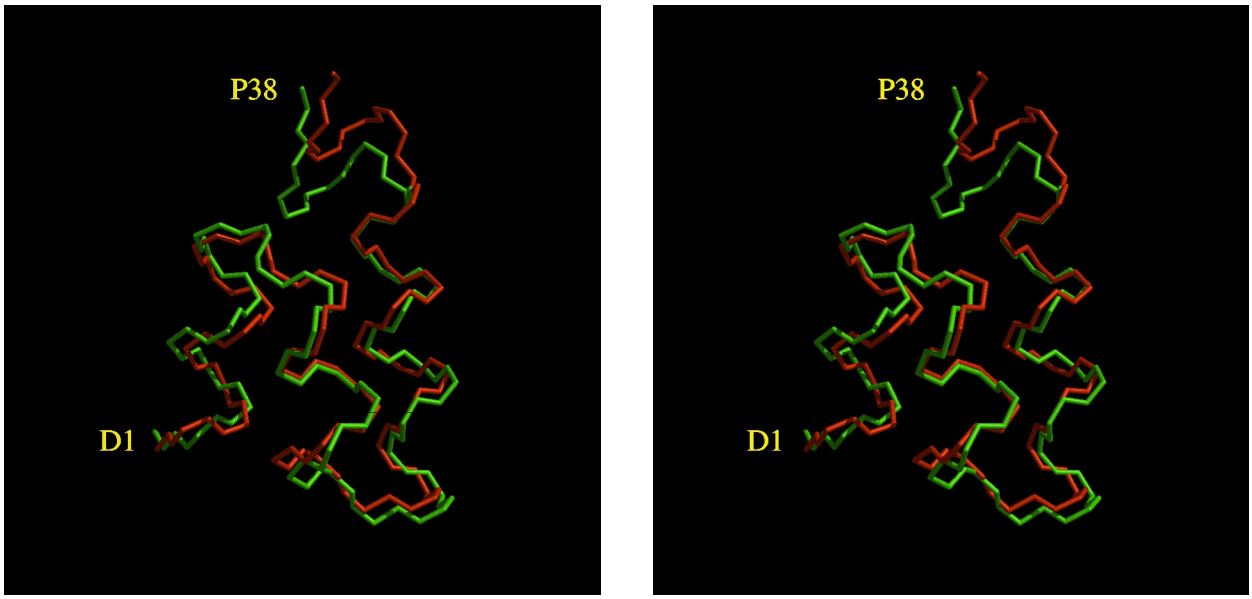


Figure 7: *The NMR structure (red) is superimposed to the structure which reached the lowest energy when the 10 unrefined NMR structures were minimized with a highly weighted protein-solvent interaction (green). Neither the NMR constraints nor the information on the disulfide bridges were used during the minimization. The largest changes occurred in the C-terminal region of the protein which is stabilized by a disulfide bridge in the native structure. The picture was prepared with the program MidasPlus* [*33*].

Results of the energy minimizations.

Figure 6 illustrates the effects of the minimizations on the structures. The structures, minimized with the ECEPP/2 energy alone, did not change towards the native structure, even though their energy values dropped considerably. In contrast, all structures minimized with the protein-solvent interaction significantly improved their rmsd values compared to the NMR reference structure. These values, which initially ranged from 6.6 Å to 11.3 Å, dropped to 4.5 Å to 7.8 Å.

A drastic difference can also be observed in the energy minimizations of 10 unrefined NMR structures. In the minimization with the protein-solvent interaction all 10 structures stayed near the native structure with a maximal rmsd value of 1.8 Å. The most significant changes occurred in the C-terminal loop which is fixed by a disulfide bridge in the native structure (Fig. 7). In the *vacuo* minimization two structures partially unfolded to rmsd values above 4 Å.

We did not expect a correlation between the final energies of the structures calculated with the protein-solvent interaction and the rmsd values (Fig.6B). The energy surface of a protein contains myriads of local minima. Even in small polypeptides the local minima of more than 2 kcal/mol above the global energy minimum show a highly complex dependence between the energy and the rmsd value, as we have shown in an exhaustive study of local minima in Met-enkephalin [18]. However, we observed that the energy refined NMR structures which have lower rmsd values, also have significantly lower energy values.

Results of the Monte Carlo simulations.

The Monte Carlo simulations with the adaptive temperature schedule [18,29] produced structures which resemble the native *Er*-10 structure. The three structures with the lowest energies have the correct three-helix topology and small rmsd values of 4.7 Å, 3.8 Å and 3.0 Å (Table 2). The quality of these three structures is illustrated in Fig.8. The comparison of the rmsd values before and after the simulation shows that all structures improved their accuracy.

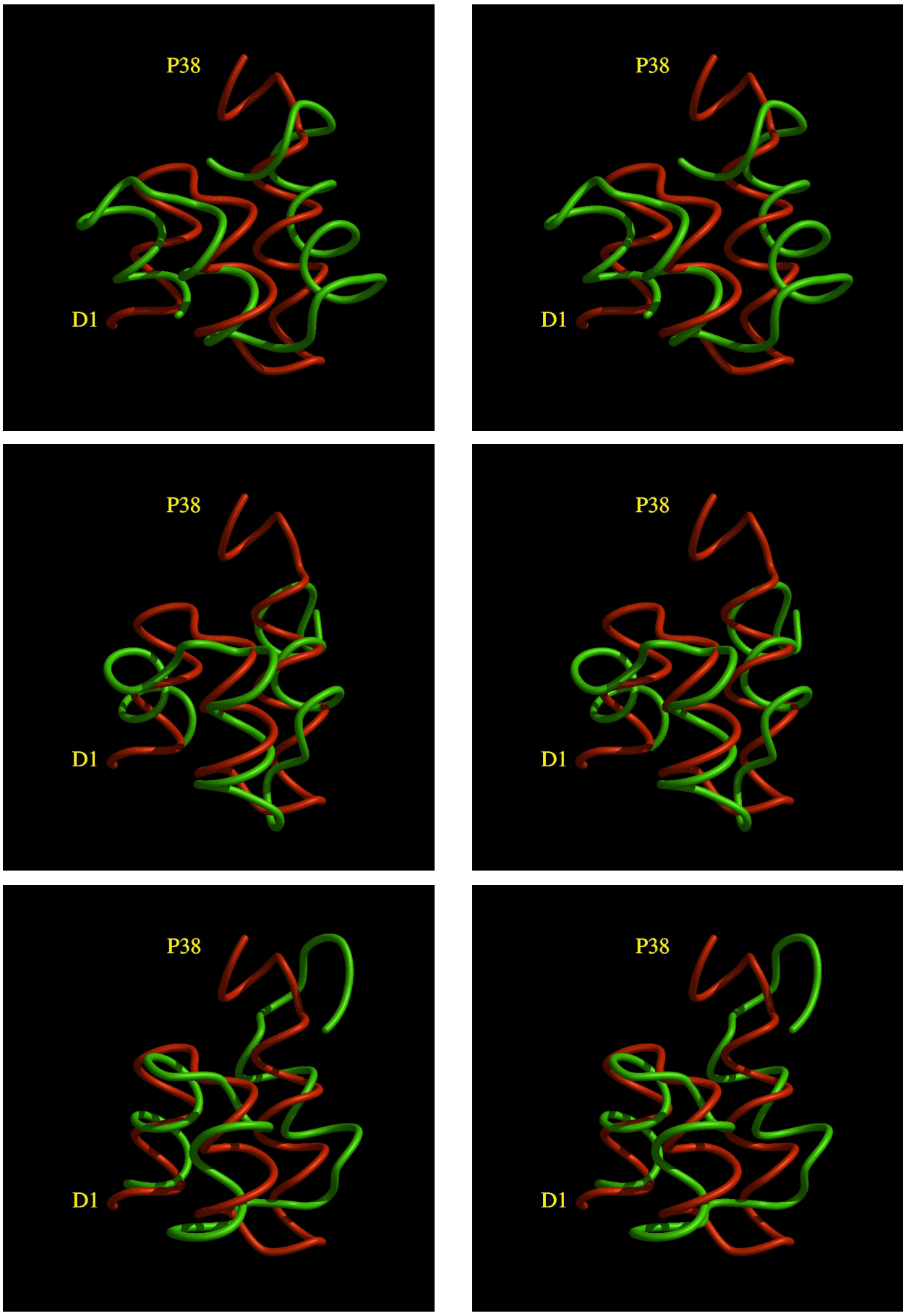
The lowest energy reached in the Monte Carlo simulations was -121 kcal mol^{-1}. In contrast, the energy minimizations starting from the three NMR structures yielded structures with energies less than -160 kcal mol^{-1}. This significant gap is mostly due to the much lower Lennard-Jones energies of the NMR structures. The energy differences between structures with the correct fold and other compact structures

Figure 8 (next page): *The three Monte Carlo structures (green) with the energies (A) -121 kcal/mol, (B) -94 kcal/mol and (C) -93 kcal/mol of Table 2 are superimposed with the NMR structure (red) in the helix regions. All three Monte Carlo structures have the correct three-helix bundle topology. The picture was prepared with the program MidasPlus [33].*

Table 2: *A comparison of structures obtained by Monte Carlo simulations and energy minimized NMR structures*

Energy [a] [kcal/mol]						Rmsd [b] [Å]	
Total	**Elect.**	**H-bond**	**Lenn.**	**Solv.**	**Torsn.**	**Start**	**End**
Monte Carlo structures:							
-121	45	-47	-218	46	53	6.9	4.7
-94	30	-46	-186	52	56	11.5	3.9
-93	22	-48	-180	46	66	8.6	3.0
-91	56	-45	-198	48	47	7.8	6.8
-84	55	-41	-192	43	51	8.8	5.5
-82	66	-47	-189	47	40	11.3	7.0
-72	69	-46	-192	41	56	7.6	5.1
-66	88	-42	-211	46	53	6.6	5.0
-52	96	-46	-197	45	50	7.1	5.4
-45	68	-44	-182	46	66	10.2	7.1
Minimized NMR structures:							
-172	51	-53	-246	42	35	0.3	1.0
-170	53	-48	-247	43	30	0.5	1.3
-162	54	-50	-244	41	36	0.3	0.7

[*a*] *Standard ECEPP/2 energies (electric, hydrogen-bond, Lennard-Jones and torsion energies) plus protein-solvent interaction energy.* [*b*] *Root-mean-square deviations measured for all backbone atoms located in the three helices compared to the NMR reference structure.*


P38
D1
P38
D1
P38
D1
P38
D1
P38
D1
P38
D1

are more subtle, e.g. between the third and fourth structure in Table 2. In that respect E*r*-10 might not be a simple test protein, as it is known that it has a large solvent exposed apolar surface area [26]. There is a major difficulty in obtaining structures with rmsd values below 2 Å when starting from unfolded structures. The problem seems to be to pack the residue side-chains correctly in the hydrophobic core. We may have to incorporate specific algorithms [23,31] into our method to overcome this problem.

Conclusions

We have shown that energy minimizations and Monte Carlo simulations with highly weighted protein-solvent interactions can fold the three helices of E*r*-10 from an unfolded state to the native state. The correct topology can be identified through the lowest total energy values including a protein-solvent interaction derived from standard estimates of the hydrophobic effect. The total energy function clearly favours the native fold. This result is not a simple consequence of compactness, as even with given three helical segments there exist many different compact folds.

We cannot justify the high weight of the protein-solvent interaction used in the first step of our calculations from experimental calorimetric data [6]. Compared to a three-dimensional profile method [32], which is based on pure statistical observations in native protein structures, or exponential decaying potential functions for the hydrophobic interactions [24], our method captures the observed dependency of the hydrophobic effect from the accessible surface area [6]. It has two major advantages: it removes low energy local minima for unfolded structures and therefore drives the structures towards a folded state and it favours burying of nonpolar side chains.

Acknowledgments
We acknowledge financial support to Ch.M. by the ETHZ. We thank Dr. C.H. Schein for critical reading of the manuscript. The use of the Cray Y-MP and the Intel Paragon parallel computer of the ETHZ is gratefully acknowledged.

References:

1. Anfinson, C.B. *Science* **1973**, *181*, 223.
2. Momany, F.A.;Mc Guire, R.F.;Burgess, A.W.; Scheraga, H.A. *J. Phys. Chem.* **1975**, *79*, 2361.
3. Weiner, P.K.; Kollmann, P.A.; Nguyen, D.T.; Case, D.A. *J. Comp. Chem.* **1986**, *7*, 230.
4. Brooks, B.R.; Bruccoleri, R.E.; Olafson, B.D.; States, D.J.; Swaminathan, S.; Karplus, M. *J. Comp. Chem.* **1983**, *4*, 187.
5. Némethy, G.; Pottle, M.S.; Scheraga, H.A. *J. Phys. Chem.* **1983**, *87*, 1883.
6. Makhatadze, G.I.; Privalov, P.L. *J. Mol. Biol.* **1993**, *232*, 639.
7. Novotny, J.; Bruccoleri, R.; Karplus, M. *J. Mol. Biol.* **1984**, *177*, 787.
8. Eisenberg, D.; McLachlan, A.D. *Nature* **1986**, *316*, 199.
9. Ooi, T.; Oobatake, M.; Némethy, G.; Scheraga, H.A. *Proc. Natl. Acad. Sci. USA* **1987**, *84*, 3084.
10. Vila, J.; Williams, R.L.; Vasquez, M.; Scheraga, H.A. *Proteins*, **1991**, *10*, 199.
11. Conolly, M.L. J. *Appl. Cryst.* **1983**, *16*, 548.
12. Richmond, T.J. *J. Mol. Biol.* **1984**, *178*, 63.
13. Shrake, A.; Rupley, J.A. *J. Mol. Biol.* **1973**, *79*, 351.
14. Perrot, G.; Cheng, B.; Gibson, K.D.; Vila, J.; Palmer, K.A.; Nayeem, A.; Maigret, B; Scheraga, H.A. *J. Comp. Chem.* **1992**, *13*, 1.
15. Wesson, L.; Eisenberg, D. *Prot. Sci.* **1992**, *1*, 227.
16. von Freyberg, B. ; Braun, W. *J. Comp. Chem.* **1993**, *14*, 510.
17. Schaumann, Th.; Braun, W.; Wüthrich, K. *Biopolymers* **1990**, *29*, 679.
18. von Freyberg, B.; Braun, W. *J. Comp. Chem.* **1991**, *12*, 1065.
19. Totrov, M.; Abagyan, R. *J. Comp. Chem.* **1994**, *15*, 1105.
20. Williams, R.L.; Vila, J.; Perrot, G.; Scheraga, H.A. *Proteins* **1992**, *14*, 110.
21. von Freyberg, B.; Richmond, T.J.; Braun, W. *J. Mol. Biol.* **1993**, *233*, 275.
22. Liwo, A.; Pincus, M.R.; Wawak, R.J.; Rackovsky, S.; Scheraga, H.A. *Protein Sci.* **1993**, *2*, 1715.
23. Tuffery, P.; Lavery, R. *Proteins* **1993**, *15*, 413-425.
24. Callaway, D.J.E. *Proteins* **1994**, *20*, 124.
25. Upon request, the source code and documentation of FANTOM 3.5 is available from the authors. Send requests to E-mail: braun@mol.biol.ethz.ch.
26. Brown, L.R.; Mronga, S.; Bradshaw, R.A.; Ortenzi, C.; Luporini, P.; Wüthrich, K. *J. Mol. Biol.* **1993**, *231*, 800.
27. Bernstein, F.C.; Koetzle, T.F.; Williams, G.J.B.; Meyer, E.F. Jr.; Brice, M.D.; Rogers, J.R.; Kennard, O.; Shimanouchi, T.; Tasumi, M. *J. Mol. Biol.* **1977**, *112*, 535.
28. Güntert, P.; Braun, W.; Wüthrich, K. *J. Mol. Biol.* **1991**, *217*, 517.
29. Li, Z.; Scheraga, H.A. *Proc. Natl. Acad. Sci USA* **1987**, *84*, 6611.
30. Chothia, C. *Nature* **1974**, *248*, 338.
31. Abagyan, R.; Totrov, M. *J. Mol. Biol.* **1994**, *235*, 983.
32. Zhang, K.Y.I.; Eisenberg, D. *Protein Sci.* **1994**, *3*, 687.
33. Ferrin, T.E.; Conrad, C.H.; Laurie, E.J.; Langridge, R. *J. Mol. Graphics* **1988**, *6*, 13.