Available online at www.sciencedirect.com
ScienceDirect
Journal of Structural Biology 158 (2007) 267–281
Journal of Structural Biology
www.elsevier.com/locate/yjsbi

Continuous representations of proteins: Construction of coordinate models from curvature profiles

A.C. Hausrath [a,*], A. Goriely [b]

[a] Department of Biochemistry and Molecular Biophysics, University of Arizona, 1041 E. Lowell, Tucson, AZ 85721, USA
[b] Department of Mathematics, Program in Applied Mathematics and BIO5 Institute, University of Arizona, USA

Received 15 April 2006; received in revised form 27 October 2006; accepted 7 November 2006
Available online 19 November 2006

Abstract

A representation of proteins based on the geometry of space curves is described. This representation enables the application of continuum methods to the analysis of macromolecular structure and form that cannot be applied to the more familiar discrete atomic coordinate models. It is shown that the continuous modeling method defines the geometry of the protein fold very efficiently. An analytical solution for curve construction is employed from which both continuous and coordinate models can be obtained. The method is applied to five representative test proteins which are used to assess the accuracy and efficiency of the modeling procedure.

Keywords: Curvature; Torsion; Polyhelix; Protein folds; Protein modeling

1. Introduction

The rapid and accelerating pace of biological structure determination has created an enormous body of data which we are only beginning to be able to interpret and utilize (Burley and Bonanno, 2002; Todd et al., 2005). As the structural biology community moves ever closer to assembling a complete catalog of protein folds used in nature, the emphasis on determining new structures must necessarily shift towards efforts to understand and organize the existing body of structural knowledge. New methods for the analysis and comparison of protein structures are needed in order to make the fullest use of this important resource.

Towards this goal, we have developed methods for the representation of proteins in terms of continuous curves, so as to be able to employ the powerful analytical apparatus applicable to continuous objects in the study of protein structure. Continuous representations are complementary to the classical atomic coordinate representation, and enable smooth deformations to study variations within and relationships between protein architectures. Many properties of proteins can only be understood from a detailed knowledge of their molecular structure, but paradoxically the discrete nature of coordinate models precludes continuous analyses which might be useful in the investigation of these properties. The goal of this work is to be able to interconvert easily between continuous and discrete representations so as to be able to study a problem from both viewpoints.

A curve-based representation of a family of α-helical repeat proteins was employed previously to examine the relationships between structurally diverse examples in this family (Hausrath and Goriely, 2006). This analysis utilized a class of spatially repetitive curves and introduced a means of constructing backbone atomic models from such curves. Here we demonstrate applicability of the method to the description of the more general case of globular α-helical proteins. The method for constructing backbone models is extended to include sidechains. These procedures enable direct conversion from the continuous representation to the discrete representation. In contrast, the conversion from a

* Corresponding author. Fax: +1 520 626 9204.
E-mail address: hausrath@email.arizona.edu (A.C. Hausrath).

1047-8477/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jsb.2006.11.003

discrete to a continuous representation utilizes an iterative fitting procedure. Recommendations on strategies for carrying out this step are described.

2. Methods

Protein folds can be represented as space curves. These curves are geometric objects distinct from the particular atomic models which display realizations of that fold. Using such continuous representations, the general description of curves possible with differential geometry can be applied to the study of protein architecture. In this manuscript we utilize an especially simple type of curve, the polyhelix, which has convenient properties for the representation of α-helical proteins. This polyhelix-based continuous representation of proteins permits an exact solution to the equations required for curve construction, and the solution simultaneously provides the coordinate systems in which to place atoms in the corresponding coordinate model.

2.1. Curve construction

A sufficiently smooth three-dimensional space curve $\mathbf{r}(s)$ can be specified in terms of the local geometric parameters curvature (κ) and torsion (τ), which define the local bending and twisting of the curve at each value of arc length s. At every point on the curve we define an orthonormal coordinate system, the Frenet frame, which has components $\mathbf{t}(s)$, $\mathbf{n}(s)$, and $\mathbf{b}(s)$, the tangent, normal, and binormal to the curve (Struik, 1988). These vectors can be obtained by differentiating the curve $\mathbf{r}(s)$ with respect to s:

$$\mathbf{t}(s) = \frac{\mathbf{r}'}{\|\mathbf{r}\|}, \quad \mathbf{b}(s) = \frac{\mathbf{r}' \times \mathbf{r}''}{\|\mathbf{r} \times \mathbf{r}''\|}, \quad \mathbf{n}(s) = \mathbf{b} \times \mathbf{t}. \tag{1}$$

More formally, curvature and torsion describe the instantaneous rate of rotation of the Frenet frame about $\mathbf{b}(s)$ and $\mathbf{t}(s)$, respectively, as the origin of the coordinate system moves along the curve. The components of the frame satisfy the Frenet equations (Gray, 1998)

$$\frac{\partial}{\partial s}\begin{pmatrix}\mathbf{t}\\ \mathbf{n}\\ \mathbf{b}\end{pmatrix} = \begin{pmatrix}0 & \kappa & 0\\ -\kappa & 0 & \tau\\ 0 & -\tau & 0\end{pmatrix}\begin{pmatrix}\mathbf{t}\\ \mathbf{n}\\ \mathbf{b}\end{pmatrix}. \tag{2}$$

The Frenet equations describe the evolution of the Frenet frame vector componentwise. Since the Frenet frame vectors can be obtained from the curve by a process of differentiation, the curve itself can be obtained by a process of integration. Curve construction is accomplished by the subsequent integration of the tangent vector $\mathbf{t}$ after solution of the Frenet equations. The complete system for evolution of the components of the Frenet frame vectors and the coordinates of the curve is

$$\begin{pmatrix}t_1\\ n_1\\ b_1\\ t_2\\ n_2\\ b_2\\ t_3\\ n_3\\ b_3\\ r_1\\ r_2\\ r_3\end{pmatrix}' = \begin{pmatrix}0 & \kappa & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ -\kappa & 0 & \tau & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & -\tau & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & \kappa & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & -\kappa & 0 & \tau & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & -\tau & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \kappa & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & -\kappa & 0 & \tau & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\tau & 0 & 0 & 0 & 0\\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0\end{pmatrix}\begin{pmatrix}t_1\\ n_1\\ b_1\\ t_2\\ n_2\\ b_2\\ t_3\\ n_3\\ b_3\\ r_1\\ r_2\\ r_3\end{pmatrix}. \tag{3}$$

2.2. Polyhelices

A solution to Eq. (3) provides not only the spatial coordinates of the curve but also the components of the vectors of the Frenet frame. These are defined in terms of the curvature and torsion profiles $\kappa(s)$ and $\tau(s)$. For general profiles, numerical integration is necessary. However, an exact analytical solution can be obtained for curvature profiles which are piecewise constant. As a curve with constant curvature and torsion is a helix, the choice of piecewise constant curvature profiles creates curves which are piecewise helical. Such curves will be referred to as polyhelices.

Piecewise constant curvature profiles can be specified by a list of ordered triples $\{\kappa_i, \tau_i, L_i\}$, each defining a segment of the polyhelix. Within each segment, the matrix in Eq. (2) is a constant matrix and the solution can be obtained by matrix exponentiation. The solution provides an exact formula for the components of the Frenet frame and the curve itself in terms of the vector $\mathbf{Y}(s) = \{t_1, n_1, b_1, t_2, n_2, b_2, t_3, n_3, b_3, r_1, r_2, r_3\}$. The derivation of the solution as a matrix product was described previously and we simply restate it here in terms of the matrix $A(\kappa, \tau, L)$ (Hausrath and Goriely, 2006).

The complete solution for any value of s is

$$Y(s) = A(\kappa_i, \tau_i; s - s_0^{(i)}) \prod_{k=i-1}^{1} A(\kappa_k, \tau_k, L_k) Y_0,$$
$$s \in \left[s_0^{(i)}, s_0^{(i+1)}\right] \quad \text{where } s_0^{(i)} = \sum_{k=1}^{i-1} L_k. \tag{4}$$

The curve is specified by curvatures only up to rotation and translation, but the initial basis vector Y_0 fixes the position and orientation in space. A parametric expression for the resulting polyhelical curve can be written in terms of the last three components of the vector $Y(s)$:

$$\mathbf{r}_{\text{ph}}(s) = (Y_{10}(s), Y_{11}(s), Y_{12}(s)) \tag{5}$$

2.3. Local Frenet frames

In addition to the coordinates of the curve, the solution in Eq. 4 provides the components of the Frenet frame at

every value of s. Parametric expressions for the tangent, normal, and binormal vectors of a polyhelix are

$$\begin{aligned}\mathbf{t}_{\mathrm{ph}}(s) &= (Y_1(s), Y_4(s), Y_7(s))\\ \mathbf{n}_{\mathrm{ph}}(s) &= (Y_2(s), Y_5(s), Y_8(s)) \text{ for } s \in \left[s_0^{(i+1)}, s_0^{(i)}\right]\\ \mathbf{b}_{\mathrm{ph}}(s) &= (Y_3(s), Y_6(s), Y_9(s))\end{aligned} \tag{6}$$

At each position s on the curve $\mathbf{r}_{\mathrm{ph}}(s)$, the Frenet vectors define a local orthogonal coordinate system $\mathbf{F}_{\mathrm{ph}}(s) = \{\mathbf{t}_{\mathrm{ph}}(s), \mathbf{n}_{\mathrm{ph}}(s), \mathbf{b}_{\mathrm{ph}}(s)\}$. A point $\{a_1, a_2, a_3\}$ in the local coordinate system $\mathbf{F}_{\mathrm{ph}}(s)$ represents a point in space at

$$\mathbf{P} = \mathbf{r}_{\mathrm{ph}}(s) + a_1\mathbf{t}_{\mathrm{ph}}(s) + a_2\mathbf{n}_{\mathrm{ph}}(s) + a_3\mathbf{b}_{\mathrm{ph}}(s) \tag{7}$$

Conversely, a point at position $\mathbf{P}$ in the external coordinates can be expressed in the local coordinate system $\mathbf{F}_{\mathrm{ph}}(s)$ by the triple $\{a_1, a_2, a_3\}$

$$a_1 = (\mathbf{P} - \mathbf{r}_{\mathrm{ph}}(s)) \cdot \mathbf{t}_{\mathrm{ph}}(s) \tag{8}$$

$$a_2 = (\mathbf{P} - \mathbf{r}_{\mathrm{ph}}(s)) \cdot \mathbf{n}_{\mathrm{ph}}(s), \tag{9}$$

$$a_3 = (\mathbf{P} - \mathbf{r}_{\mathrm{ph}}(s)) \cdot \mathbf{b}_{\mathrm{ph}}(s). \tag{10}$$

2.4. Residue frames: local frames at C_α positions

Atomic coordinate models are typically expressed in an orthogonal cartesian coordinate system, but other coordinate systems may be useful in certain situations. One example is crystallographic coordinates, which are the appropriate coordinates for applying space group symmetry operations. For a curve-based description of protein structure, the local coordinate systems on the curve are a natural choice in which to express the atomic coordinates. In particular, for curves which pass through the C_α positions of protein models, the coordinate systems located at these points are convenient choices in which to express the coordinates of the atoms of the corresponding residue so as to allow construction of coordinate models of proteins from curvature profiles.

For polyhelical curves defined by a list of curvature triples $\{(\kappa_i, \tau_i, L_i)\}$ that represent a protein fold, the values of s at which to locate residue-specific coordinate systems can be obtained from the curve by requiring that the spatial separation of the points on the curve correspond to the standard C_α–C_α distance, the length of a peptide plane, or 3.8 Å. The separation in arc length between two such points varies with κ_i and τ_i. An example is shown in Fig. 1.

Within a segment specified by a triple $\{\kappa, \tau, L\}$ the distance between two points on the curve separated by an arc length s can be obtained using Eq. 4. Taking $Y_0 = \{1,0,0,0,1,0,0,0,1,0,0,0\}$ the vector to the point on the curve at an arc length s is

$$P_s = \left\{\frac{\sin(s\alpha)\kappa^2 + s\tau^2\alpha}{\alpha^3}, \frac{\kappa - \kappa\cos(s\alpha)}{\alpha^2}, \frac{\kappa\tau(s\alpha - \sin(s\alpha))}{\alpha^3}\right\} \tag{11}$$

and so the spatial separation of two points separated by arc length Δs is

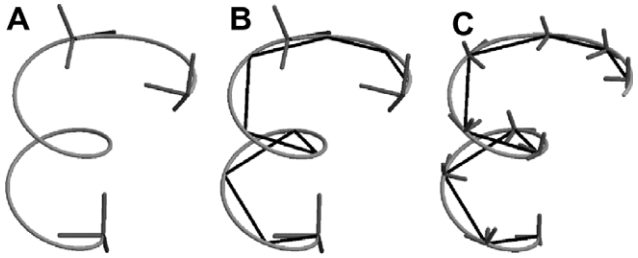

Fig. 1. Location of C_α positions along curve consisting of multiple segments. (A) A 2-segment polyhelix. The Frenet frames at the endpoints of the segments are shown, with $\mathbf{t}_{\mathrm{ph}}$, $\mathbf{n}_{\mathrm{ph}}$, and $\mathbf{b}_{\mathrm{ph}}$ in blue, green, and red, respectively. Note these frames are used in curve construction but are not associated with any residue. (B) Trace of C_α positions (black) located at spatial separations of 3.8 Å along the curve. Note that the C_α positions 7 and 8 cross a segment boundary. (C) Residue frames (in gray) are shown at these C_α positions. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$$d(\Delta s) = \sqrt{\frac{\Delta s^2\tau^4 + \kappa^2(\Delta s^2\tau^2 + 2) - 2\kappa^2\cos(\Delta s\alpha)}{\alpha^4}} \tag{12}$$

Determination of the value of Δs needed to achieve a particular spatial separation d_{sep} can be accomplished with standard nonlinear fitting routines such as Newton's method. This value is then used for all the points within

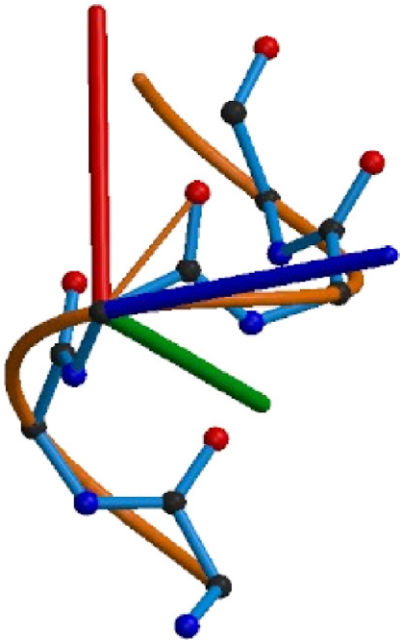

Fig. 2. Construction of atomic models in local coordinate systems. A residue frame, with the tangent vector in blue, normal vector in green, and binormal vector in red is shown on the curve, in orange. A backbone model constructed from the curve is shown, consisting of two residues on either side of the local frame itself. Atoms are constructed in the individual frames using the local coordinates. An example vector is displayed from the origin of the local frame to the carbonyl oxygen of the associated residue. This atom is located at local coordinates $\{a_1, a_2, a_3\} = \{1.4468, 0.4592, 1.8605\}$ in the local Frenet frames consisting of the tangent, normal, and binormal vectors. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

the segment. For example, within an α-helical segment with $\kappa = 0.38$ and $\tau = 0.15$, the value 4.23 is used.

More generally, an initial guess Δs_{init} for input to a numerical optimization algorithm can be obtained by expanding the $\cos(\Delta s\alpha)$ term in the expression for $d(\Delta s)$ and solving for Δs:

$$\Delta s_{\text{init}} = \frac{\sqrt{2}}{\kappa}\sqrt{3 - \sqrt{9 - 3\kappa^2 d_{\text{sep}}}} \tag{13}$$

If two successive points lie on different segments, the spatial separation of points depends on the curvatures of several segments (see Fig. 1B). The arc length between the points has contributions from segments in which the speed of the parametrization is not identical. In principle an analytical expression could be obtained but in practice a stepwise nonlinear optimization which makes use of Eq. 12 to evaluate the distance has proved satisfactory. Therefore, the set of arc length values $\{s_i\}$ at which to locate the C_α atoms and the local coordinate systems associated with the corresponding residues from a list of curvature triples can be obtained with the following steps:

(1) Determine Δs_j to use within each segment j by nonlinear optimization.
(2) Starting at the beginning of segment 1 with $s = 0$, while $s_{i+1} < s_0^{(j+1)}$, obtain $s_{i+1} = s_i + \Delta s_j$.
(3) Obtain Δs_{cross} to cross the boundary between segments with nonlinear optimization and define $s_{i+1} = s_i + \Delta s_{\text{cross}}$.
(4) Repeat steps (2) and (3) for each segment j.

The set of $\{s_i\}$ also defines the set of residue frames $\{\mathbf{F}_i\} = \{\mathbf{F}_{\text{ph}}(s_i)\}$. These residue frames are the natural choice for the coordinate system in which to express the atomic coordinates of a model.

2.5. Local coordinates of atoms in residue frames

The origin of each $\mathbf{F}_i$ defines the location of the C_α atom and so the local coordinates of each C_α with respect to the local basis $\mathbf{F}_i$ are $\{0,0,0\}$. The advantage of the local coordinate description lies in use of the approximation that, up to choice of rotamers, amino acids have the same local coordinates. The approximation is strictly true only within α-helical segments of ideal geometry. Outside such regions, although the resulting models can have distorted peptide planes in the turn regions, appropriate selection of curvature parameters can reduce this effect, and the use of this assumption leads to considerable simplification in the representation of protein models. In this section the local coordinates of the amino acids appropriate for use in polyhelix-based models are obtained. Section 2.5 describes how these local coordinates can be used to create atomic models, and Section 3 explores the extent of validity of the assumption by comparison of the resulting models defined by curvatures with experimentally determined coordinates.

The strategy used here for obtaining local coordinates is to create coordinate models of α-helices with ideal geometry for which a parametric curve passing very near to the C_α positions can be obtained. The accuracy of the curve is essential for the accurate determination of the local coordinates.

The parametric expression for a helix oriented with their axes along z used to fit the coordinate models is

$$h(R,\omega,P,s) = (R\cos(\omega s), R\sin(\omega s), P\omega s) \tag{14}$$

where R is the helix radius, $2\pi P$ is the pitch (the z-distance between two points on the helix with equal x and y). The variable ω is adjusted so that s is an arc length, that is

$$\omega = \pm\frac{1}{\sqrt{P^2 + R^2}} \tag{15}$$

which implies that the length of the curve per turn is $l = 2\pi/\omega$. By adjusting R and P this expression can be made to coincide with the coordinates of a supplied helical model.

The program EDPDB (Zhang and Matthews, 1995) was used to create coordinate models of idealized α-helices with Ramachandran angles of $\phi = -57.0$ and $\psi = -47.0$ oriented with its axis along z. A helical model was constructed for each rotamer in the rotamer library of Lovell et al. (2000). Each model consists of 100 residues. Fitting of the parametric expression in Eq. (14) to the α-carbon coordinates of the resulting models gave estimates of the parameters for R, P, and ω (Table 1). From these values and Eq. (14) the values for curvature and torsion were obtained using the helical relations Tables 2 and 3.

Table 1
Idealized α-helix parameters

Parameters	Values
R	2.2748
P	0.8903
ω	0.4094
κ	0.3812
τ	0.1492

Values of the geometric parameters radius (R), pitch (P), frequency (ω), curvature (κ), and torsion (τ) used for an α-helix of idealized form.

Table 2
Mean coordinate error

Model	C_α	Backbone	Sidechain
1ENH	1.20	1.19	1.69
1NA3	1.13	1.09	1.85
1BRR	1.77	1.77	2.16
2LZM	2.02	2.30	2.77
1YPI	1.26	1.37	2.09
ALL	1.55	1.64	2.20

The average residual error by category for the different models shows the general trend that the C_α models and backbone atoms are quite comparable in terms of accuracy, with an average error on the order of 1.6 Å. The error on the sidechain atoms are slightly larger on average, with an average error of 2.2 Å.

Table 3
Parameter utilization

Model	Residues	C_α	Backbone	Complete	$\kappa\tau$	Ratio
1ENH	46	138	552	1239	21	0.152
1NA3	82	246	984	2118	39	0.159
1BRR	216	648	2592	5085	75	0.116
2LZM	162	486	1944	3927	132	0.271
1YPI	241	723	2892	5541	240	0.332

The table gives the number of residues for each model and the number of parameters used for the C_α, backbone, and complete models. For comparison, the number of parameters required to specify the curvature profile used is given in the column headed $\kappa\tau$. The ratio of the number of parameters for the coordinate and curvature descriptions of the C_α model is in the "Ratio" column. Additional parameters are used in the sidechain-containing models (the coordinates in the distinct rotamers) complicating the comparisons of the actual number of parameters required for a curvature-based representation. Nevertheless these are determined independently of the particular model under consideration, and only the curvature parameters and the identity of the rotamers are specific to a given model.

$$\kappa_{\text{helix}} = \frac{R}{P^2 + R^2} \quad \text{and} \quad \tau_{\text{helix}} = \frac{P}{P^2 + R^2\omega^2} \tag{16}$$

From the parametric representation of the curve, the local frames at the locations of the C_α positions were constructed with Eqs. (5) and (6). The coordinates for the model expressed in external coordinates were converted to local coordinates with Eqs. (8)–(10). The values for the local coordinates at all 100 positions in each model were averaged to minimize any differences due to numerical errors in the fitting or model construction process. The resulting sets of local coordinates for the rotamers are included in the Supplementary material.

2.6. Construction of atomic models

Employing the local coordinate descriptions of amino acids (Eq. 7) in conjunction with the polyhelix solution (Eq. 4) for curve construction enables complete atomic models to be built. The information required consists of a list of curvature triples which defines the fold of the model, and a list of amino acids with the rotamer specified to be placed on the model derived from that fold.

For each residue, the appropriate arc-length value s_i provides the residue frame $\mathbf{F}_i$. In this frame the local coordinates $\{a_1, a_2, a_3\}$ for each atom from the specified rotamer is converted to external coordinates using Eq. 7 as illustrated in Fig. 2.

Construction proceeds in the following stages:

(1) Construct polyhelix curve from curvature profile specified by curvature triples $\{\kappa_i, \tau_i, L_i\}$.
(2) Determine arc length values $\{s_i\}$ on the curve at which to locate the residue frames $\{\mathbf{F}_i\}$.
(3) Construct the residue frames $\{\mathbf{F}_i\}$ with Eq. 6.
(4) For each residue construct the atoms in the local frame with local coordinates from Section 2.5.
(5) Convert the local coordinates to external coordinates with Eq. 7.
(6) Repeat steps (4) and (5) for all residues.

2.7. Procedure for devising curvature profiles

The methods above enable construction of models from curvature profiles in a deterministic way. The reverse process, determining curvature profiles from coordinate models, is achieved by an iterative optimization process. There are infinitely many distinct curves that pass through a given set of points and so there are infinitely many different possible curvature profiles. The question then becomes one of finding from all of these possibilities a particular solution which has useful properties. Using the polyhelix approximation is especially appropriate in this context because these curves can be exactly specified with a discrete set of parameters, which drastically simplifies the search. But there is no general result on the number of helical arcs needed to fit a given set of points with a specified accuracy. Therefore, we have used an empirical approach to assign the number of arcs, increasing the number as needed to achieve the desired accuracy.

The basic problem of quantifying agreement between a discrete point set and a continuous curve does not have a unique or exact solution. Our strategy has been to work with point sets derived from the curve, which are more easily compared to experimentally derived atomic coordinates. Curvature profiles described here were obtained by minimizing a target function which is the sum of distances (referred to as the residual error) between successive points in the experimentally derived model and a superimposed curve-derived model. The target function is

$$TF(P) = \sum_i \|\mathbf{r}(s_i) - \mathbf{x}_i\| \tag{17}$$

where $\mathbf{r}(s_i)$ are the curve-derived C_α positions and $\mathbf{x}_i$ are the C_α positions of the target model. The list $P = \{(\kappa_i, \tau_i, L_i)\}$ contains the set of triples specifying the curvature profile.

Formulated in this way, the problem is a multiparameter minimization using the components of P. Many algorithms exist for treating these types of problems (Press, 1992). But optimization schemes can be very sensitive to both the mathematical form of the target function and the initial conditions used to start the optimization. We are presently exploring different approaches to test their effectiveness for curvature profile optimization. At present no substantive advantage to any particular method has been identified. The main difficulty appears to be in devising suitable starting conditions for optimization, and this is independent of the particular optimization scheme employed subsequently. In fitting the curvature profiles described here we have employed the very simple algorithm steepest descent (Press, 1992).

Our implementation of the steepest descent algorithm approximates the partial derivatives of the target function with respect to parameter P_n (which can be a curvature, torsion, or length from any segment of the list P) using the difference quotient

$$\frac{\Delta TF(P)}{\Delta p_n} = \frac{TF(P_n) - TF(P)}{\delta} \tag{18}$$

where P_n is a modified list P in which the parameter P_n has been increased by the small quantity δ and all other parameters are the same as in P. These are used to construct the gradient to identify the modifications to the parameters which will accomplish the most rapid decrease in the target function. The values of δ found to be most effective for curvature and torsion optimization were in the range 0.00005–0.005. For optimizing segment length parameters values in the range 0.01–0.1 were typically used.

The problem of assigning the initial conditions is simplified for α-helical proteins because the helices themselves are easily recognized and assigned known curvature values. Our approach has been to parse the structures into overlapping structural fragments starting and ending with successive helices in the model under consideration. These structure fragments were fitted by alternating manual adjustment with automatic minimization of the residual error.

The initial estimate employed for the length of the helical segments is the number of residues in the α-helix multiplied by 4.23 (obtained from Eq. 12), and the curvature values for these segments is fixed at $\kappa = 0.3800$ and $\tau = 0.1500$. Initial assignment of the curvature values for the inter-helix portions of the fragments were made by visual inspection of models of the helix–loop–helix segments using Xtalview (McRee, 1999). Starting curvature estimates were used to create curve-derived C_α coordinate models as described in Section 2.4. These curvature values were adjusted manually, new models recalculated, and the results compared by visual inspection to the target structural segments. The number and lengths of segments used to model the portion of the fragment between the helices was also assigned by visual inspection and depends on its length and complexity. When suitable candidate profiles had been devised by hand, these were subjected to steepest descent to improve the fit.

Because of the difficulty of defining the orientation of the initial basis in irregular portions of a protein structure, in some examples residues near the chain termini were not included in the curve-based models. To define the initial basis $\mathbf{Y}_0$ for each selected fragment, an idealized curve-derived α-helix model located at the origin of the external coordinate system was superimposed on the initial α-helix of the fragment using the method of Kabsch (Kabsch, 1976). The resulting transformation was applied to a parametrically defined helical curve with $\kappa = 0.3800$ and $\tau = 0.1500$ also located at the origin, and the initial Frenet basis $\mathbf{Y}_0$ used in the subsequent fitting process was obtained as the tangent, normal, and binormal vectors at the initial point of this transformed helix.

Using this initial basis and the initial estimates of the curvature profile parameters for each segment, polyhelix models for the fragment were constructed. Optimization by steepest descent was employed to decrease the residual error. When the residual error stopped decreasing significantly, small manual adjustments to the curvature parameters were made based on visual inspections using Xtalview and then automatic optimization reapplied.

When the structure fragments showed no further improvement by either manual or automatic fitting, they were joined together into longer submodels, which were themselves subjected to the same iterative fitting/optimization procedure. Eventually all the fragments were joined together for a final refinement process to create the overall curvature profiles.

2.8. Pitfalls in the curvature profile optimization process

The basic strategy of an iterative refinement method is to make small changes in the model parameters, assess their effect on the residual error, and update the parameters so as to decrease the error. Some characteristics of the optimization problem which cause difficulties in achieving a high-quality model are described here and where possible strategies to overcome, avoid, or ameliorate these are suggested.

The optimization of curvature profiles using the polyhelix construction is frequently stalled by entrapment within local minima. To avoid this, it proved useful to divide the parameters into multiple sets which were optimized independently, intermittently alternating between these sets. When optimization using one set no longer resulted in improvements, switching to the other set could allow progress to resume as the parameters would be in a local minimum with respect to one set but not with respect to the other. This strategy also proved to be faster than optimizing all the variables at once.

A characteristic of the polyhelix optimization is that parameters for segments near the beginning of the curvature profile affect most of the model, but parameters for segments late in the curvature profile influence only a smaller portion of the model. Therefore, different parameters have unequal influence on the residual error, and consequently optimization is less effective at improving parameters in the latter part of the profile. A related point is that small changes in parameters early in the profile can result in large changes in portions of the model remote from the region of the structure described by these parameters due to the lever-arm action of the intervening portions of the model. In extreme cases the small steps needed for this reason could go below the limit of numerical precision. The converse of this effect is that parameters early in the profile can be very responsive to optimization, but this responsiveness requires that the model for the latter part of the profile be rather accurate. Therefore, at the outset of optimization, when the model is not especially accurate, improvement algorithms tend to be less effective overall, and more manual intervention is needed. As the model improves, the traction of the algorithms improves. To exploit this property, it is useful to optimize models for the fragments separately, so that when they are

joined the latter portions of the model are as accurate as possible.

However, when joining fragment models, the initial basis for a fragment is that from the end of the previous fragment. Usually the orientation of that basis, as defined by the curvature profile up that point, does not optimally orient the later fragment. Therefore, some adjustments in the previous fragment model must be made to accommodate the new fragment, making the model more accurate overall but possibly at the cost of decreasing the quality of the fit locally in the previous fragment. Since some of the effort involved in making the individual fragment models accurate is undone by this process, a balance between time spent in fitting the individual fragment models and fitting the multi-fragment models must be sought.

In some situations the process gets stuck, and neither manual or automatic adjustments are able to improve a model which visually is clearly not optimal. In such situations it is always possible to divide one or more curvature profile segments so as to introduce additional degrees of freedom into the optimization, and this can usually break the impasse so that the refinement can proceed. For example, many proteins have bent or kinked helices, whereas we have restrained the α-helical curvature segments to idealized geometry and so the optimization procedure cannot respond to these structural variations. In the early stages of fitting these distortions may not be as significant as other errors but as the model improves it can become limiting. Breaking a segment into multiple parts can allow the model to conform to these deviations from ideality. Note that in this case the degrees of freedom can reduce error not only within the α-helical segments but can lead to more favorable orientations of subsequent portions of the model.

Another source of error which stalls the fitting algorithms is when accumulated errors in the lengths of the segments build up so that the points of the curve-derived model are out of phase with the target coordinate model. In helical portions of the structure the atoms may be nearly superimposed because minimizing the residual error will tend to align the axes of the helical portions of the models even if the atoms nearest in space do not correspond. The best strategy is to avoid this outcome for when it happens it usually cannot be repaired. The lengths of segments with nonzero curvatures have a strong influence on the form of the curve model, and it is not usually possible to disentangle the errors in the lengths. The best course is to restart the fitting process at the point where the correspondence between atoms in the models begins to diverge.

2.9. Example of curvature-based model optimization

Optimizing a model by manipulating its curvature profile has some characteristics which may be useful in various contexts in structural biology. Modest changes in curvature profiles can result in gross changes in form of a structural model. A consequence is that in some cases a curvature profile may be within the radius of convergence of a curvature-based optimization algorithm while the starting model itself is so inaccurate so as to be outside the radius of convergence of a coordinate-based algorithm.

An example is shown in Fig. 3 where the final curvature profile for 1ENH was perturbed and then 20 cycles of optimization performed with the steepest descent algorithm. In Fig. 4 the progress of the fitting procedure, by cycle, is shown for the four tests. The perturbations of the curvature parameters were fairly modest (magnitude 0.1). In this example, three of the four initial profiles appear to lie within the radius of convergence of the algorithm, while the fourth does not. By visual inspection, there is no obvious distinction between these models which would have enabled this to be predicted. Similarly, when making initial curvature estimates in practice, it is often necessary to try a variety of initial conditions when fitting a given portion of the model. The *rmsd* values for the starting structures and the target varied between 8.1 and 11.1 Å for the three successful runs. In the fourth case where a local minimum was encountered the starting *rmsd* was 10.6 Å. Therefore, the *spatial* accuracy of the starting model is not necessarily the best predictor of the eventual success of a curvature-based optimization scheme.

2.10. Test cases

To assess the efficiency and accuracy of the methods described above for constructing coordinate models from curvature profiles, we consider five test cases for which curvature profiles have been determined. The accuracy of the curve-derived model depends strongly on the quality of the curvature profile used. Here the intent is to demonstrate that, with sufficiently accurate curvature profiles, complete and accurate models coordinate models can be constructed. The test cases are

(a) Engrailed homeodomain: 3-helix bundle (PDB code 1ENH, residues 9–54) (Clarke et al., 1994).
(b) CTPR2: idealized TPR repeat, 5 helices (PDB code 1NA3, residues 1–81) (Main et al., 2003).
(c) Bacteriorhodopsin: 7-helix bundle (PDB code 1BRR, residues 10–225) (Essen et al., 1998).
(d) T4 Lysozyme: globular (PDB code 2LZM, residues 2–163) (Weaver and Matthews, 1987).
(e) Triosephosphate isomerase: globular (PDB code 1YPI, residues 4–243) (Lolis et al., 1990).

For each test case, a model including sidechains was constructed. At each residue, the best-fitting sidechain rotamer was selected by systematically constructing all the different rotamers for the appropriate amino acid at that site into the residue frames as described in Section 2.5. These sidechain models were ranked by calculating the residual error on the sidechain atoms for each constructed rotamer. The rotamer selected for inclusion in the final model was the one with the smallest residual error.

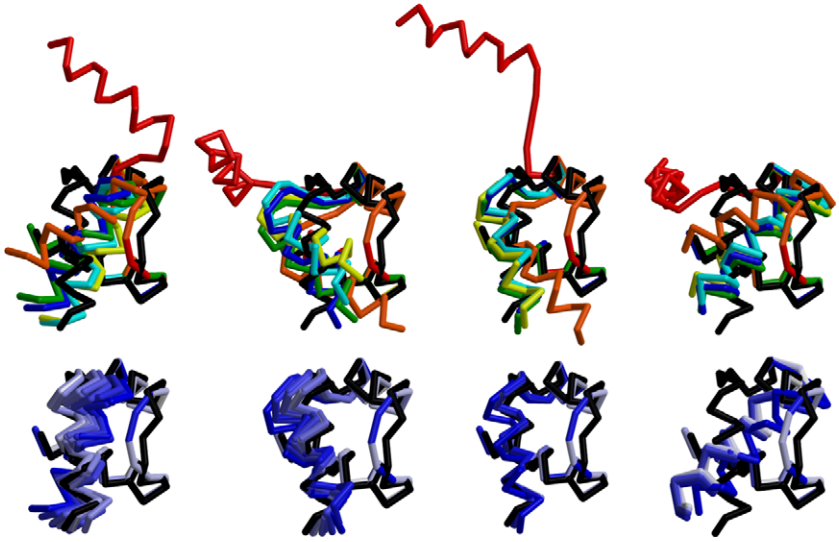

Fig. 3. Example runs of fitting algorithm. Example optimization runs 1–4 (from left to right) starting with perturbed models of 1ENH. In the top row, the starting conformation is displayed in red and the models resulting from the first five cycles are displayed in orange, yellow, green, cyan, and blue successively. In the bottom row are models resulting from cycles 5 to 20, fading from blue (cycle 5) to white (cycle 20). In each case the experimentally determined C_α trace is shown in black. In the last run, a local minimum is encountered at cycle 6. The other three runs converge to the unperturbed solution after different numbers of cycles. In this example only the curvature parameters from segments 5 and 6 were perturbed. Denoting the changes in curvature and torsion for segment n as $\Delta\kappa_n$ and $\Delta\tau_n$, the four examples employed the modifications $\Delta\kappa_5 = +0.1$, $\Delta\tau_5 = -0.1$, $\Delta\kappa_6 = +0.1$, $\Delta\tau_6 = -0.1$ (run 1); $\Delta\kappa_5 = -0.1$, $\Delta\tau_5 = +0.1$, $\Delta\kappa_6 = -0.1$, $\Delta\tau_6 = +0.1$ (run 2); $\Delta\kappa_5 = +0.1$, $\Delta\tau_5 = +0.1$, $\Delta\kappa_6 = -0.1$, $\Delta\tau_6 = -0.1$ (run 3); and $\Delta\kappa_5 = -0.1$, $\Delta\tau_5 = +0.1$, $\Delta\kappa_6 = +0.1$, $\Delta\,\tau_6 = -0.1$ (run 4). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

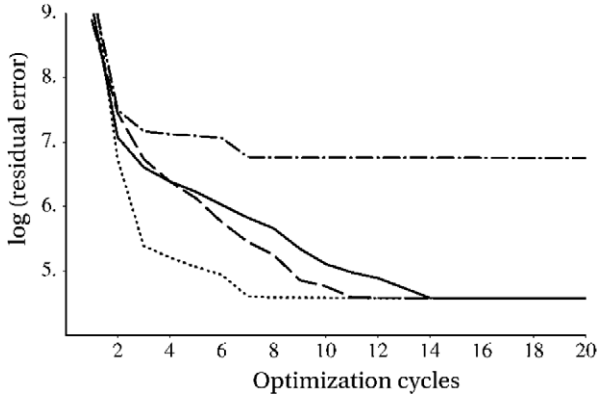

Fig. 4. Progress of fitting runs. Graphs of logarithm of residual error for the fitting runs displayed in Fig. 3. The individual traces represent run 1 (solid line), run 2 (dashed), run 3 (dotted) and run 4 (dot-dash). The first three runs converge to the unperturbed values.

Fig. 5 shows the curvature profile used, the curve specified by this profile, and the superposition of the experimentally determined C_α trace on the C_α trace obtained from the curve for each of the five test cases. The parameters defining the curvature profiles are tabulated in Supplementary material.

3. Discussion

Proteins are inherently complicated molecules and any model must necessarily include a significant number of parameters. On the other hand, sufficient experimentally determined information to simultaneously and independently determine all of these parameters is not always obtainable. It is, therefore, crucial to devise modeling methods which can be tuned so as to make optimal use of the available information. Continuous methods offer the prospect of structural descriptions which have this property. The curvature-based representation described here can be adapted so as to employ the minimum information necessary to describe an accurate protein model, and therefore, the method is a first step toward such a goal. The five curve-derived models constitute a test of this approach.

Models with increasing information content are more able to express finer and finer structural details, and the high-resolution coordinate models obtained by X-ray crystallography and solution NMR represent one extreme. However, a variety of experimental methods exist which provide some structural information but which at present may not provide sufficient information to constrain a detailed atomic coordinate model: examples include solid-state NMR, single-particle reconstruction by electron microscopy, fiber diffraction, and small-angle X-ray scattering. However, in many cases these techniques are applicable to molecules which are inaccessible to the higher-resolution methodologies. We suggest that curvature-based models may be especially useful in this context.

Specification of a protein model using curvatures is a purely geometric description. It is equivalent to a coordinate description, but the spatial form of the model is encoded in a different way, using different variables. It is possible to construct physical models using these curvature variables, just as it is possible to define physical interactions

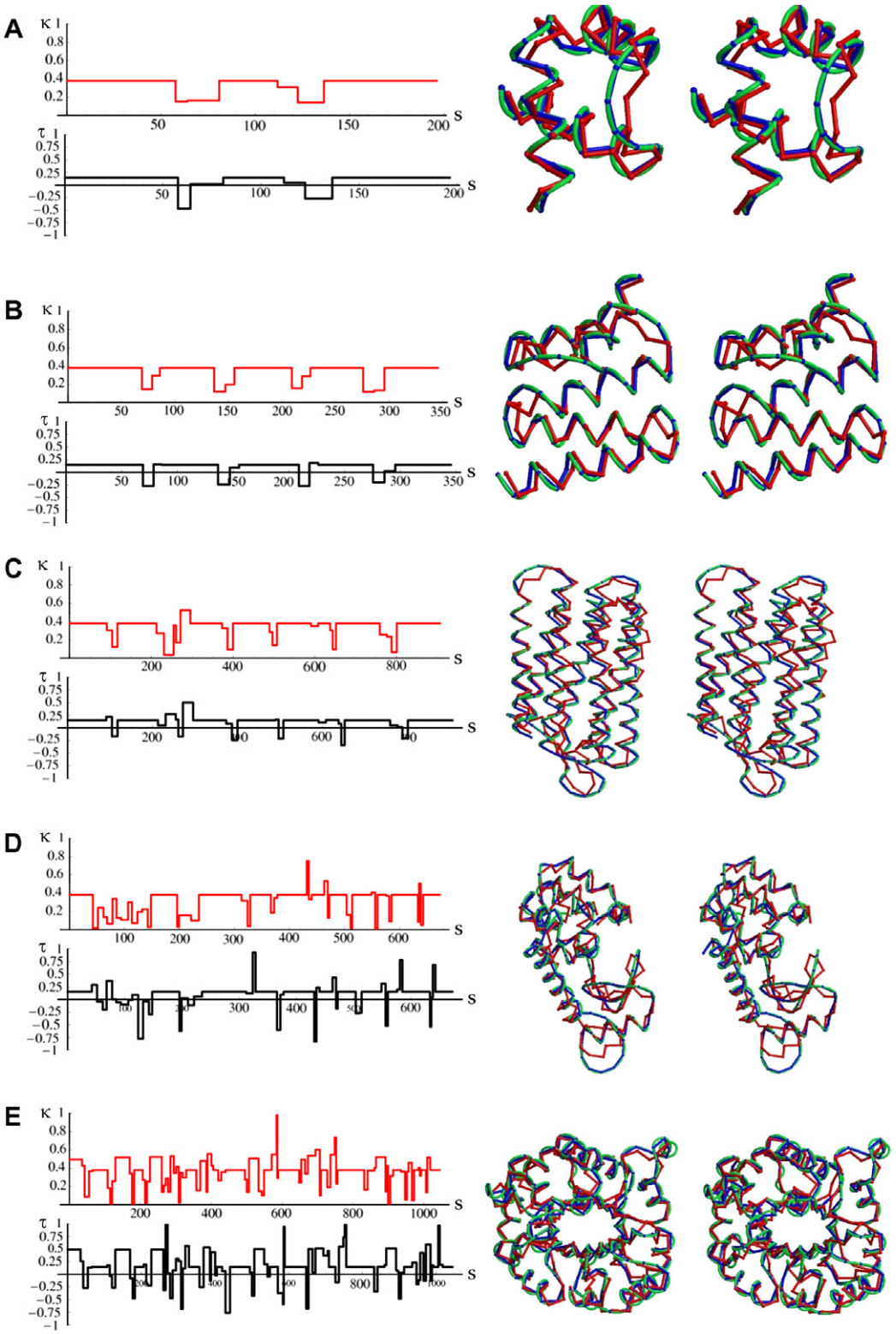

Fig. 5. Comparison of coordinate and curve representations of models. (A) Left: curvature (red) and torsion (black) profiles. Right: stereo representations of experimental C_α trace (red), curve representation (green) and curve-derived C_α trace (blue). The models are 1ENH (A); 1NA3 (B); 1BRR (C); 2LZM (D); and 1YPI (E). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

in terms of coordinate variables, and thereby use coordinate models as the starting point for physical theories about molecular properties. Some physical interaction energies are naturally expressed in terms of spatial variables, but there may be classes of models which could make use of energies defined partly or wholly in terms of curvature

variables. It is important to realize that any such physical models must start with assumptions about such energies, whereas the geometric curvatures themselves, which specify the structure, are more fundamental and exist independently of any such physical assumptions.

To assess the strengths and weaknesses of a curvature-based representation of proteins, the accuracy of the models has been determined independently for the C_α positions, for the backbone atoms, and the sidechain atoms. Increases in the accuracy of the model can always be achieved by introducing additional parameters, but a point is reached where a large number of additional parameters may be needed to make a modest improvement in model accuracy. To place these observations on a quantitative basis, the subsequent discussion describes first the accuracy of the models, and then the amount of information required to specify them.

3.1. Model accuracy

In this manuscript we do not consider the possibility of coordinate errors in the experimental structures, and attempt only to approximate the coordinate sets with curve-derived models. However, to assess the applicability to experimental structural biology it will be necessary to compare curve models directly with experimental data.

Fig. 6 shows the distribution of errors in each category for the five models as a general assessment of this modeling method. For these molecules it is possible to obtain C_α and backbone coordinate models with better than 2 Å average coordinate error. Sidechain coordinates have slightly higher average errors, between 2 and 3 Å average coordinate error for the test cases used here.

The distribution of coordinate errors in the five fitted models is not uniform, nor are they randomly distributed. Histograms of the average error in the C_α coordinates for each of the models are shown in Fig. 6. In each case, the distributions have a peak suggesting that a substantial fraction of the molecule is modeled at a similar, relatively high level of accuracy. But some fraction of the model is less accurately modeled corresponding to the tails in the distributions of errors.

By examining the coordinate error along the chain in Fig. 7, it is clear that the helical regions (shaded) are more accurately modeled, and so the tails in the distributions in Fig. 6 are mostly contributed by the portion of the model outside the helical regions. The disparity is illustrated in the separate error histograms of the helical and nonhelical regions (Fig. 8). The histogram for the nonhelical regions is much broader. Note that, therefore, the more accurate parts of the model are also those parts which require the least amount of information to describe. The representation has the property that it can accurately depict a substantial fraction of the model with a minimal amount of information.

The accuracy of the helical regions is greater, and is limited by the placement of the α-helical segments of the curve. In contrast in the turn regions the construction method can distort the peptide planes and tetrahedral geometry of the α-carbons, and so both the accuracy of the curvature profile and the steric distortion can contribute to positional error in the backbone atoms. Use of appropriate values of the curvatures in the turn regions can reduce these effects but will not remove them entirely.

Despite the simplifying assumptions, the method is quite effective. The models superimpose accurately on the exper-

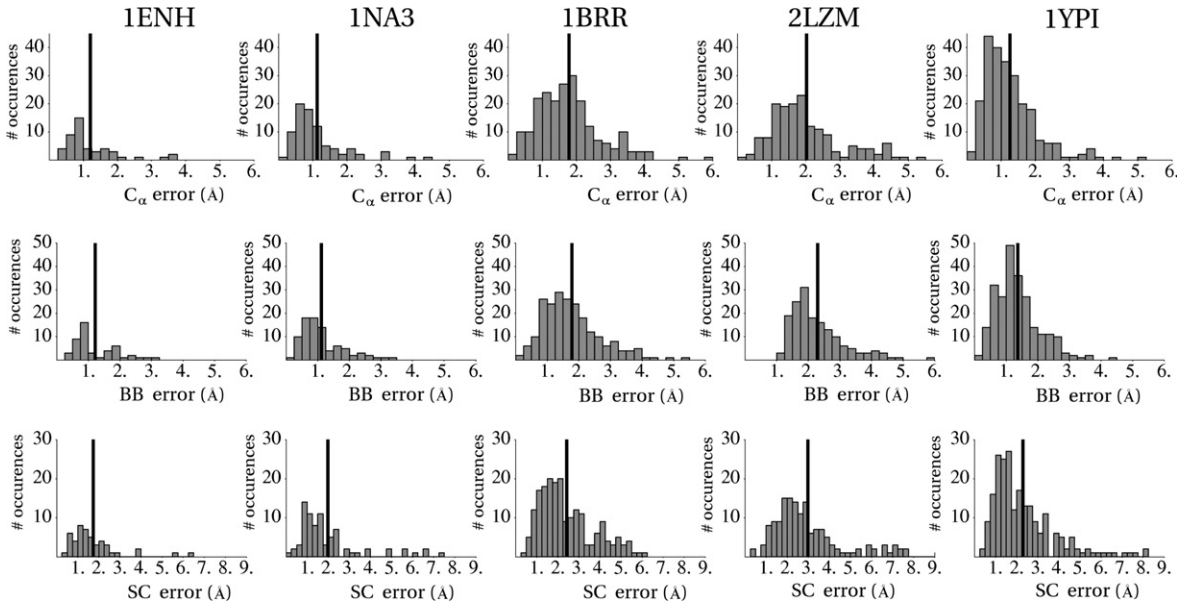

Fig. 6. Error histograms showing the distribution of error values in the individual models (by column), broken down into error on C_α positions (top row), average error on backbone atoms (middle row), and average error on sidechain atoms (bottom row). The solid vertical lines indicate the mean value of the distributions.

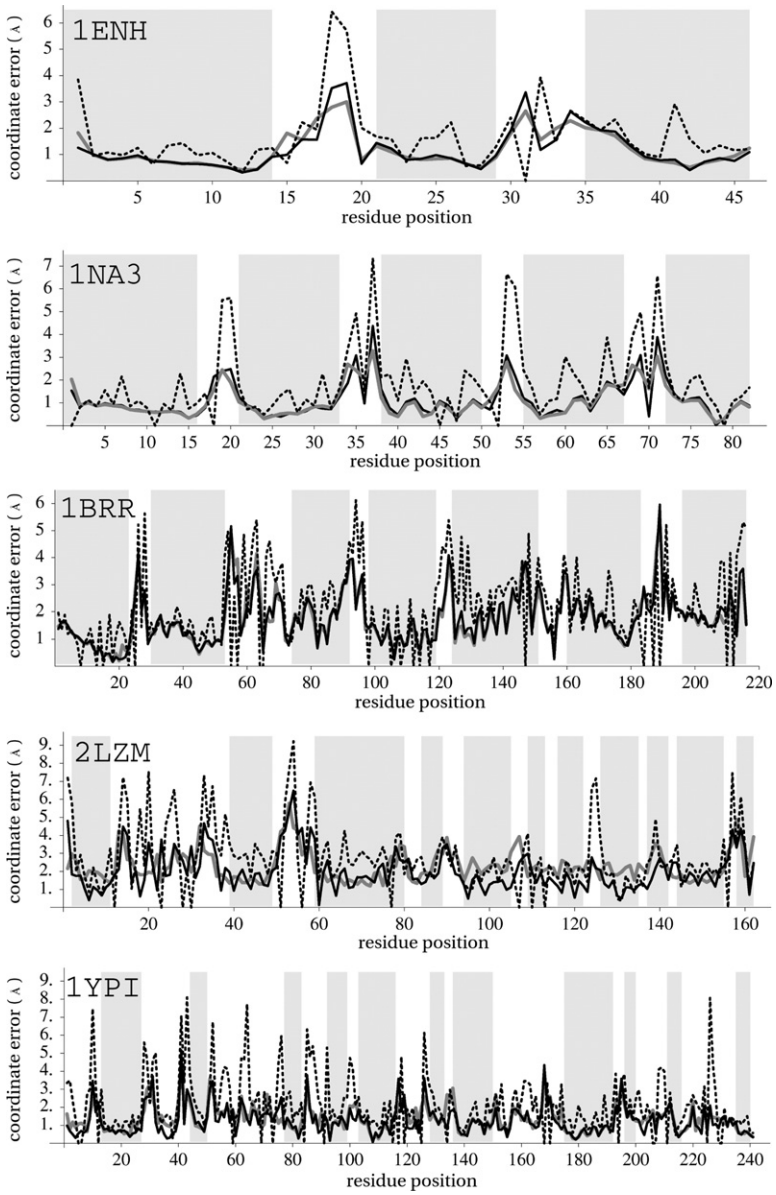

Fig. 7. Model accuracy by residue location of errors in the five models, in Angstroms. Solid black lines denote pointwise positional error of C_α positions obtained from the curve, in Angstroms. Solid gray lines denote error on backbone coordinates, and dashed lines denote average error in sidechain coordinates (set to zero for glycine residues). The α-helical regions of the sequence are shaded.

imentally determined coordinates. C_α trace superpositions for the five models are shown in Fig. 5. Graphical representations of the backbone and sidechain models superimposed on the experimentally determined structure of 1ENH are shown in Figs. 9 and 10 and show excellent agreement.

While the curve exactly specifies the C_α trace and the backbone atoms, it cannot specify the conformation of the sidechains. Additional information must be provided to fix their orientations. One strategy would be to construct a sidechain model from the backbone model alone through ranking the sidechain conformation on energetic considerations. In this study we take a simpler approach. By selecting conformations from a library of the most common sidechain rotamers (Lovell et al., 2000) and placing them in the local coordinate systems as specified by the curvature profile, we demonstrate that accurate models can be constructed. These models are the best possible models given the particular curvature profile and rotamer library used. Using this method, the model is completely specified by a curvature profile and a list of amino acid rotamers at each position.

One expects that the accuracy of the curvature profile limits the accuracy of the models derived from them. However, it is difficult to estimate the error in the values of the curvatures. Instead, we make use of the related but not identical quantity of the positional error in the C_α coordinates as a substitute. Using the five models described above, Fig. 11 shows that there is a strong correlation between the error in the C_α coordinates and both the

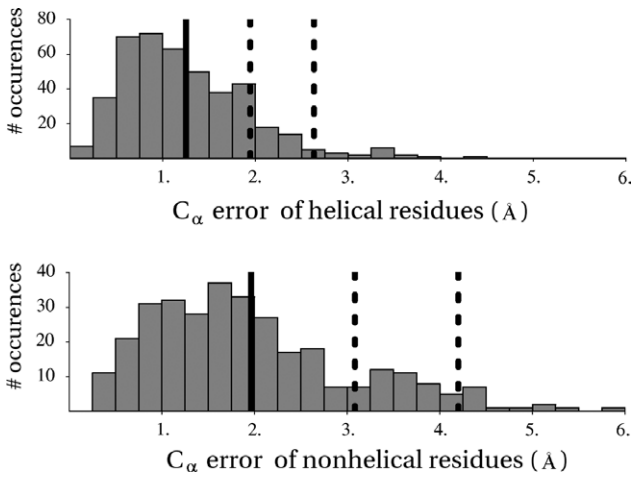

Fig. 8. Histogram of C_α positional errors in helical and nonhelical regions. Histogram of positional error in C_α coordinates for all helical residues (top) and all nonhelical residues (bottom) in the five fitted models. Mean values of the coordinate errors (1.25 and 1.96 Å, respectively) are demarcated by solid lines. One and two standard deviations (0.69 and 1.11 Å, respectively) above the mean are demarcated by dashed lines. In general the C_α positions in helical regions are more accurate and the distribution of errors in these regions is less broad.

backbone and sidechain atoms. Therefore, the first requirement for a high-quality model is an accurate curvature profile.

It is clear that overall the helical regions are more accurately modeled by this method than the turn and sheet regions. The proteins chosen here are primarily composed of α-helical structure and so the accuracy of the method is greater for these. A larger proportion of these models is comprised of the type of structure for which the method is well-suited. Improved methods for obtaining accurate curvature profiles is likely to result first in improvements in these regions.

A substantial part of the aggregate error in the models lies outside the α-helical regions. And as the curvature profile becomes more and more optimal, this tendency is reinforced and the residual error is concentrated outside the helical regions. Improvements in the curvature profile will not address this problem. Therefore, it will be important to devise improved ways to model the turn and sheet regions.

3.2. Model efficiency

A means to assess the effectiveness of the representation which takes into account both the efficiency of parameter usage and the accuracy of the model which it achieves simultaneously is described in this section. It enables comparison between structurally distinct models and also provides a guideline for incorporation of additional segments during model construction.

Some proteins have more complex structures than others, and require more parameters to achieve an accurate representation. For example, in the test cases, bacteriorhodopsin (1BRR) is a larger protein than T4 lysozyme (2LZM), but has a simpler structure and this is reflected in the fact that 75 parameters were found sufficient to achieve an adequate fit to 1BRR while 132 parameters were used to fit 2LZM. Comparing these models on a per-residue basis, the 1BRR model used 75 parameters for 216 residues, or about 0.35 parameters/residue. The 2LZM model used 132 parameters for 162 residues, or about 0.81 parameters/residue. Therefore, the efficiency of the 1BRR model is greater. But despite the fact that the 1BRR model was more efficient in parameter usage, it also has a slightly lower average coordinate error (1.78 Å on the C_α positions compared to 2.02 Å for 2LZM). So both in terms of efficiency and in terms of accuracy, the method is more successful with the 1BRR model.

In general, improvements in model accuracy can always be achieved by using more segments, which require more curvature parameters and therefore, decreases the efficiency. When a model is inaccurate, a few additional degrees of freedom may result in a significant improvement in model accuracy. However, if a model is already fairly accurate, a large number of additional parameters might be required to achieve significant improvement. Because some proteins are more complex than others, an accurate description of their structure must inherently contain a larger amount of information than for others. From this point of view, an optimal curvature profile would be one which approaches this minimal information content. It is difficult to assess the "information content" of a coordinate model. But curvature profiles are a 1-dimensional specification of the information needed to construct such a 3-dimensional model. The information in them is represented linearly, and can be analyzed with the tools of signal processing. A natural quantification of the complexity of a continuous signal $f(s)$ is the Shannon entropy

$$\mathbf{S}[f] = -\int P[f(s)] \ln P[f(s)] \, \mathrm{d}s \tag{19}$$

where $P[f(s)]$ is the probability that the signal f will take the particular value observed at s. In the context of calculating entropy of a curvature profile fitted to a protein, there is no obvious choice for the expected probability distribution. The most rigorous approach would be to obtain the distribution from statistics derived from a collection of curvature profiles of fitted proteins, but these are not presently available. Here we employ a normal distribution with the same mean and standard deviation that the values of the curvature profile display.

The Shannon entropy can be considered as the "information carrying capacity." A signal with a larger entropy has the *ability* to carry more information than a simple signal. But whether that capacity is fully utilized depends on the particular instance. In the context of this discussion, a curvature profile with many segments may have a high entropy and so would be capable of specifying a very complex protein fold. But if that profile is used to specify a relative simple fold, the profile is not an efficient one.

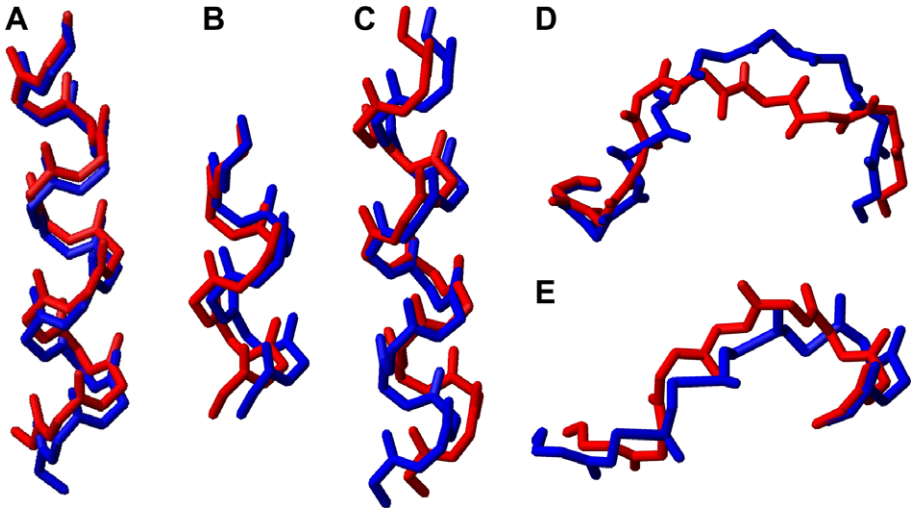

Fig. 9. Backbone comparison for 1ENH. Backbone model of residues 9–55 of structure 1ENH (red) and model specified by curvature profile (blue). (A) helix 1 (residues 9–21); (B) helix 2 (residues 29–37); (C) helix 3 (residues 43–55); (D) turn 1 (residues 22–28); and (E) turn 2 (residues 38–42). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

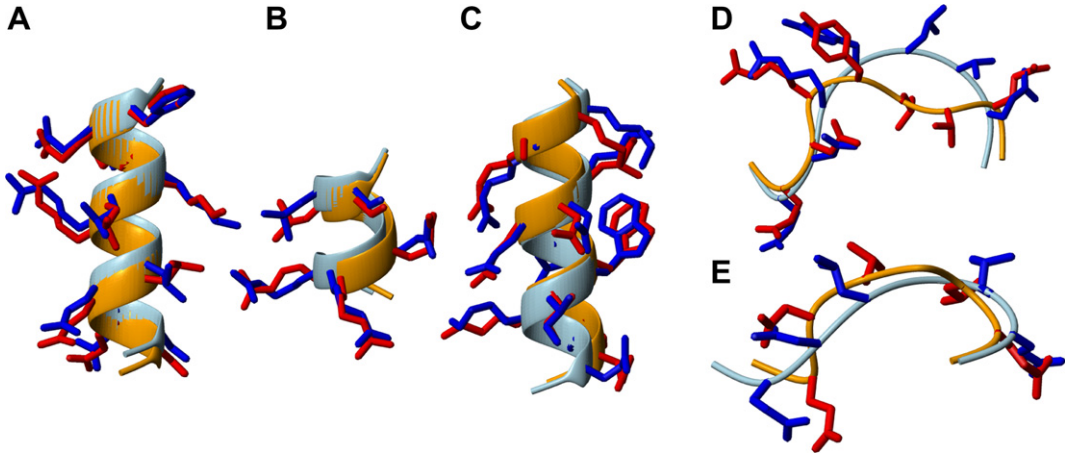

Fig. 10. Sidechain comparison for 1ENH. Sidechain models for experimental structure 1ENH (red) and model specified by curvature profile (blue). The segments are identical with those in Fig. 9. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Therefore, the most efficient models will be those with high accuracy and yet with low entropies in their curvature and torsion profiles. As a criterion for the assessment of model quality taking into account both accuracy and efficiency, we propose the benchmark quantity

$$\mathbf{Q} = (\text{average } C_\alpha \text{ coordinate error})\sqrt{\mathbf{S}[\kappa(s)]\mathbf{S}[\tau(s)]} \qquad (20)$$

The quantity **Q** has the property that accurate models have lower values, but for two equally accurate models, the model achieved with the simpler curvature description will be lower. To compare the quality of two structurally dissimilar models, the ratio of their **Q** values can be taken. This ratio has the property that, when comparing two models of approximately equal complexity, the two are judged primarily on their relative accuracy. But when comparing a complex model and a simple model, the relative accuracy is weighted by their relative complexity, reflecting the greater difficulty of achieving an accurate model for more complex structures.

Table 4 ranks the five models by their **Q** values. By this criterion, 1BRR is the most efficient of the four models, and 1YPI is the least. The 2LZM and IYPI models model have a substantially higher **Q** value than the other three. The comparison suggests that a more accurate model for T4 lysozyme and triosephosphate isomerase should be obtainable using the same number of curvature parameters or alternatively that a simpler curvature profile might be devised that results in an equally accurate model.

3.3. Applications

A curvature profile generates a curve which in turn specifies the locations of the C_α positions of a model. A variety of approximate methods exist for construction of backbone

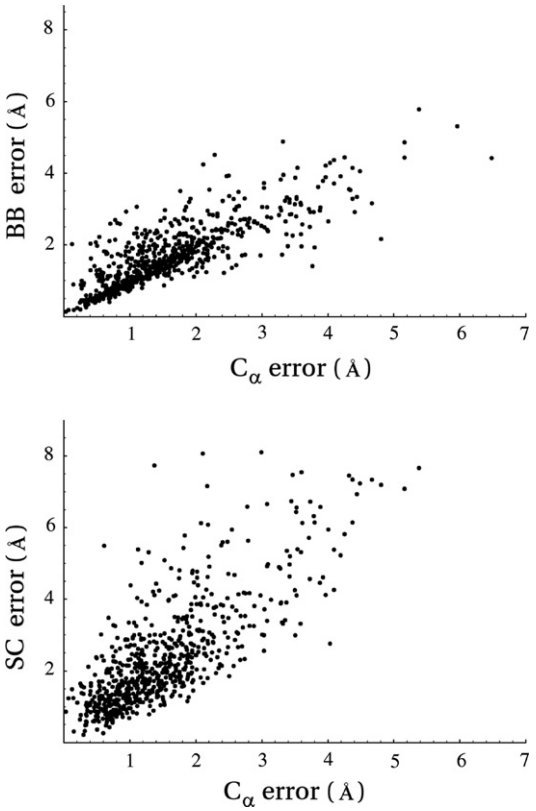

Fig. 11. Error correlations. Top: average error in backbone coordinates plotted vs. the error in C_α position for all amino acids in the five fitted models. The correlation coefficient is 0.83. Bottom: average error in sidechain coordinates plotted vs. error in the C_α position for all amino acids in the five fitted models. The correlation coefficient is 0.78.

or complete atomic coordinate models using only the positions of these atoms (Correa, 1990; DePristo et al., 2003; Luo et al., 1992; Milik et al., 1997; Payne, 1993; Purisima and Scheraga, 1984; Rey and Skolnick, 1992; Wang et al., 1998). These methods often include energy minimization or other forms of optimization, in the process making small changes in the resulting structure. This is one way which could be used to construct models from curvature profiles. An advantage of such an approach is that the geometry of the resulting models may be improved in the turn regions, but a disadvantage for some purposes is that atomic locations are no longer precisely controlled by the geometric description.

Table 4
Model efficiency as assessed by **Q** function

Model	C_α error	κ entropy	τ entropy	**Q**
1BRR	1.78	−0.110	−0.053	0.135
1NA3	1.13	−0.165	−0.093	0.140
1ENH	1.20	−0.193	−0.106	0.172
1YPI	1.26	−0.188	−0.193	0.240
2LZM	2.02	−0.216	−0.084	0.272

The three components and the value of the quantity **Q** quantifying model efficiency are displayed by column for the five fitted models.

The curve also specifies not only the C_α positions but also the backbone atoms. Various methods exist to construct sidechain models with a given sequence from a given backbone model (Ponder and Richards, 1987; Jones and Thirup, 1986; Lee and Subbiah, 1991; Desmet et al., 1992). This problem is of particular current interest in homology modeling (Kopp and Schwede, 2004). The use of a curvature profile for specifying the overall fold to which these methods could be applied has the advantage that it is possible to systematically explore "nearby" conformations or folds by manipulation of the curvature profile while applying proven methods for sidechain positioning. This capability may be especially useful for homology modeling of variable regions of proteins (Moult, 2005; Rohl et al., 2004).

A different problem arises in protein engineering and design, where the identity of some or all of the sidechains must be *determined* from a backbone model (Dahiyat and Mayo, 1997; Dwyer and Hellinga, 2004; Dantas et al., 2003). Backbone models based on curves specified by curvature profiles may provide scaffolds suitable for protein design applications using existing methods. Alternatively, it might prove useful to incorporate geometric methods into the design calculations themselves. Because of the small number of parameters required to specify a curvature profile, manipulation of the curvature profile may be an efficient way to incorporate backbone freedom into the protein design process (Mooers et al., 2003; Pokala and Handel, 2001).

Finally, some problems in structural biology could benefit by utilizing the continuous representation. The efficient description of structure in terms of relatively few parameters may enable the limited information content of some structural techniques to be used to constrain a continuous model. Another possibility would be to vary a model systematically by manipulation of its curvature profile, and score the agreement of the resulting models with the available experimental data as a means to find a structural model which best matches it.

3.4. Conclusions

Atomically detailed protein models can be constructed from curves specified by curvature profiles. In this manuscript we have employed a simple class of curvature profiles consisting of piecewise constant segments, which specify curves comprised of connected helical arcs. Both continuous curve models and discrete atomic coordinate models may be obtained from these curvature profiles using this polyhelix construction. The resulting models described here may find use in a variety of applications.

A key property of the polyhelix representation is that it is very efficient in terms of the number of parameters it requires. The method can devise models with average errors in the range of 2–3 Å, while using a fraction of the parameters required for an explicit coordinate model. In general helical regions are more accurately modeled.

Improvement in the methods for the description of turn and β-sheet regions is an area that needs further attention.

Acknowledgments

The authors gratefully acknowledge support for this work from the Bio5 Institute (A.G. and A.H.), the Department of Biochemistry and Molecular Biophysics, University of Arizona (A.H.), and the National Science Foundation. This material is based upon work supported by the National Science Foundation under Grant No. DMS-0604704 (A.G.) and NSF Grant DMS-IGMS-0623989 (A.G.).

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at doi:10.1016/j.jsb.2006.11.003.

References

Burley, S.K., Bonanno, J.B., 2002. Structuring the universe of proteins. Annual Review of Genomics and Human Genetics 3, 243–262.

Clarke, N.D., Kissinger, C.R., Desjarlais, J., Gilliland, G.L., Pabo, C.O., 1994. Structural studies of the engrailed homeodomain. Protein Science 3 (10), 1779–1787.

Correa, P.E., 1990. The building of protein structures from α-carbon coordinates. Proteins: Structure Function and Genetics 7 (4), 366–377.

Dahiyat, B.I., Mayo, S.L., 1997. De novo protein design: fully automated sequence selection. Science 278 (5335), 82–87.

Dantas, G., Kuhlman, B., Callender, D., Wong, M., Baker, D., 2003. A large scale test of computational protein design: folding and stability of nine completely redesigned globular proteins. Journal of Molecular Biology 332 (2), 449–460.

DePristo, M.A., De Bakker, P.I.W., Shetty, R.P., Blundell, T.L., 2003. Discrete restraint-based protein modeling and the C_α-trace problem. Protein Science 12 (9), 2032–2046.

Desmet, J., Demaeyer, M., Hazes, B., Lasters, I., 1992. The dead-end elimination theorem and its use in protein side-chain positioning. Nature 356 (6369), 539–542.

Dwyer, M.A., Hellinga, H.W., 2004. Periplasmic binding proteins: a versatile superfamily for protein engineering. Current Opinion in Structural Biology 14 (4), 495–504.

Essen, L.O., Siegert, R., Lehmann, W.D., Oesterhelt, D., 1998. Lipid patches in membrane protein oligomers: crystal structure of the bacteriorhodopsin-lipid complex. Proceedings of the National Academy of Sciences of the United States of America 95 (20), 11673–11678.

Gray, A., 1998. Modern Differential Geometry of Curves and Surfaces with Mathematica, second ed. CRC Press, Boca Raton.

Hausrath, A.C., Goriely, A., 2006. Repeat protein architectures predicted by a continuum representation of fold space. Protein Science 15 (4), 753–760.

Jones, T.A., Thirup, S., 1986. Using known substructures in protein model-building and crystallography. EMBO Journal 5 (4), 819–822.

Kabsch, W., 1976. Solution for best rotation to relate 2 sets of vectors. Acta Crystallographica Section A 32, 922–923.

Kopp, J., Schwede, T., 2004. Automated protein structure homology modeling: a progress report. Pharmacogenomics 5 (4), 405–416.

Lee, C., Subbiah, S., 1991. Prediction of protein side-chain conformation by packing optimization. Journal of Molecular Biology 217 (2), 373–388.

Lolis, E., Alber, T., Davenport, R.C., Rose, D., Hartman, F.C., Petsko, G.A., 1990. Structure of yeast triosephosphate isomerase at 1.9 Å resolution. Biochemistry 29 (28), 6609–6618.

Lovell, S.C., Word, J.M., Richardson, J.S., Richardson, D.C., 2000. The penultimate rotamer library. Proteins: Structure Function and Genetics 40 (3), 389–408.

Luo, Y., Jiang, X.L., Lai, L.H., Qu, C.X., Xu, X.J., Tang, Y.Q., 1992. Building protein backbones from C-alpha coordinates. Protein Engineering 5 (2), 147–150.

Main, E.R.G., Xiong, Y., Cocco, M.J., D'Andrea, L., Regan, L., 2003. Design of stable alpha-helical arrays from an idealized TPR motif. Structure 11 (5), 497–508.

McRee, D.E., 1999. Xtalview xfit—a versatile program for manipulating atomic coordinates and electron density. Journal of Structural Biology 125 (23), 156–165.

Milik, M., Kolinski, A., Skolnick, J., 1997. Algorithm for rapid reconstruction of protein backbone from alpha carbon coordinates. Journal of Computational Chemistry 18 (1), 80–85.

Mooers, B.H.M., Datta, D., Baase, W.A., Zollars, E.S., Mayo, S.L., Matthews, B.W., 2003. Repacking the core of T4 lysozyme by automated design. Journal of Molecular Biology 332 (3), 741–756.

Moult, J., 2005. A decade of CASP: progress, bottlenecks and prognosis in protein structure prediction. Current Opinion in Structural Biology 15 (3), 285–289.

Payne, P.W., 1993. Reconstruction of protein conformations from estimated positions of the C-alpha coordinates. Protein Science 2 (3), 315–324.

Pokala, N., Handel, T.M., 2001. Review: protein design—where we were, where we are, where we're going. Journal of Structural Biology 134 (2-3), 269–281.

Ponder, J.W., Richards, F.M., 1987. Tertiary templates for proteins—use of packing criteria in the enumeration of allowed sequences for different structural classes. Journal of Molecular Biology 193 (4), 775–791.

Press, W.H., 1992. Numerical Recipes in C: The Art of Scientific Computing, second ed. Cambridge University Press, Cambridge, NY.

Purisima, E.O., Scheraga, H.A., 1984. Conversion from a virtual-bond chain to a complete polypeptide backbone chain. Biopolymers 23 (7), 1207–1224.

Rey, A., Skolnick, J., 1992. Efficient algorithm for the reconstruction of a protein backbone from the alpha-carbon coordinates. Journal of Computational Chemistry 13 (4), 443–456.

Rohl, C.A., Strauss, C.E.M., Chivian, D., Baker, D., 2004. Modeling structurally variable regions in homologous proteins with rosetta. Proteins-Structure Function and Bioinformatics 55 (3), 656–677.

Struik, D.J., 1988. Lectures on Classical Differential Geometry, second ed. Dover Publications, New York.

Todd, A.E., Marsden, R.L., Thornton, J.M., Orengo, C.A., 2005. Progress of structural genomics initiatives: an analysis of solved target structures. Journal of Molecular Biology 348 (5), 1235–1260.

Wang, Y.L., Huq, H.I., de la Cruz, X.F., Lee, B.K., 1998. A new procedure for constructing peptides into a given C alpha chain. Folding and Design 3 (1), 1–10.

Weaver, L.H., Matthews, B.W., 1987. Structure of bacteriophage-T4 lysozyme refined at 1.7 Å resolution. Journal of Molecular Biology 193 (1), 189–199.

Zhang, X.J., Matthews, B.W., 1995. EDPDB—a multifunctional tool for protein-structure analysis. Journal of Applied Crystallography 28, 624–630.