Published in final edited form as:
Proteins. 2011 October ; 79(10): 2828–2843. doi:10.1002/prot.23108.

PackHelix: a tool for helix-sheet packing during protein structure prediction

Chengcheng Hu,
Department of Computer Science, University of California, Davis, CA 95616

Patrice Koehl, and
Department of Computer Science and Genome Center, University of California, Davis, CA 95616

Nelson Max*
Department of Computer Science, University of California, Davis, CA 95616

Chengcheng Hu: cechu@ucdavis.edu; Patrice Koehl: koehl@cs.ucdavis.edu; Nelson Max: max@cs.ucdavis.edu

Abstract

The three-dimensional structure of a protein is organized around the packing of its secondary structure elements. Predicting the topology and constructing the geometry of structural motifs involving α-helices and/or β-strands are therefore key steps for accurate prediction of protein structure. While many efforts have focused on how to pack helices and on how to sample exhaustively the topologies and geometries of multiple strands forming a β-sheet in a protein, there has been little progress on generating native-like packing of helices on sheets. We describe a method that can generate the packing of multiple helices on a given β-sheet for $\alpha\beta\alpha$ sandwich type protein folds. This method mines the results of a statistical analysis of the conformations of $\alpha\beta_2$ motifs in protein structures to provide input values for the geometric attributes of the packing of a helix on a sheet. It then proceeds with a geometric builder that generates multiple arrangements of the helices on the sheet of interest by sampling through these values and performing consistency checks that guarantee proper loop geometry between the helices and the strands, minimal number of collisions between the helices, and proper formation of a hydrophobic core. The method is implemented as a module of ProteinShop. Our results show that it produces structures that are within 4–6 Å RMSD of the native one, regardless of the number of helices that need to be packed, though this number may increase if the protein has several helices between two consecutive strands in the sequence that pack on the sheet formed by these two strands.

Keywords

Protein structure prediction; super secondary structure; *ab initio* modeling; helix-sheet packing

1 Introduction

The analysis of protein structures provides fundamental insights into most biochemical functions and consequently into the cause and possible treatment of many diseases. This understanding has led to the development of structural biology as a discipline aimed at studying the 3D conformation of biomolecules. It is also the rationale behind structural genomics whose aim is to determine the structures of all known proteins. This task is far from being completed, as current experimental methods for finding the structure of a protein are very time consuming and expensive. In addition they are limited in scope as they are

*Corresponding author; nlmax@ucdavis.edu.

more appropriate for small, globular proteins that only represent a biased sample of the protein universe. One way to circumvent this problem is to use computational methods to predict the structure of a protein. *In silico* protein structure prediction methods have the advantages of being faster and much less costly, while still providing models with reasonable precision [1]. Various methods have been developed to solve the protein structure prediction problem. They can be classified into two general categories, template-based structure prediction and *ab initio* structure prediction. Template-based methods have been so far the most successful, and their applications are expected to increase as the databases of known structures that can serve as templates keep on growing. However they fail for proteins for which no template can be identified; in such cases, *ab initio* methods represent the only option. As the native structure of a protein corresponds to a stable thermodynamic state with minimum energy, it is theoretically possible to reach this stage through energy minimization. Currently however, only approximations of the actual energy function that governs protein structure stability are known, and *ab initio* methods based solely on the minimization of these approximate functions have not yet been successful over a large range of proteins. The shortcomings of these methods have been attributed to the approximations in the energy function as well as to the difficulties encountered when minimizing these highly non-linear functions, that have many local minima. Current *ab initio* methods therefore usually include two main components: a sampling search mechanism that generates raw models as initial conformations, followed by a global minimization of their energies [2,3]. Our goal here is to generate these sample initial conditions.

The standard approach to sample the conformational space of a protein structure is to arrange folding motifs such as the α-helix and the β-strand in 3D space to form sample models. It is challenging to sample the interactions between these folding motifs, especially the nonlocal interactions between two strands as well as between a strand and a helix. To some extent, this problem can be circumvented by redefining the folding motifs. This is the approach developed by Taylor and colleagues, [4, 5] who predicted protein structures from ideal forms derived from “stick” models, in which motifs consisting of layers of either packed α-helices or hydrogen-bonded β-strands are arranged into a protein domain. The different ideal forms are decided by the topology that connects secondary structures in a protein, which is determined by considering secondary links in the form of intra-chain hydrogen bonds (secondary structure) and tertiary links formed by the packing of secondary structures. The table of these ideal forms defines a “periodic table”. Based on this idealized representation of motifs, Taylor *et al.* were able to generate a set of thousands of rough structural models for a given protein sequence that usually contains a fold that is close to the native structure [5]. A more general approach is to consider the secondary structure packing problem directly. The arrangement of strands within a β-sheet is constrained by different local and nonlocal interactions such as hydrogen bonds between strand pairs. As the geometry of these interactions is well understood, packing strands to form sheets is mostly a sampling problem. In that respect, Bradley and Baker [6,7] developed a folding tree structure with a multilevel sampling method, which allowed them to explicitly sample alternative strand pairings with nonlocal connections while still considering all possible local interaction between strand pairs. In parallel, Max *et al.* [8] developed a tool, BuildBeta, as a module in the ProteinShop system [9], which, given enough space and time, creates all possible arrangements into β-sheets of β-strands derived from secondary structure predictions. BuildBeta orders the possible strand topologies by the scores of Ruczinski [10] so that the most probable can be considered first, and selects for each of these topologies several favorable strand alignments using the alignment scores developed by Zhu and Braun [11]. Using this approach, Max *et al.* managed to generate structures that are within 4–6Å RMSD of the corresponding native ones, even for some large barrel-shaped proteins of 300 residues.

In contrast to the well-defined hydrogen bonds that govern strand-strand interactions, the interactions between helices and β-sheets involve mostly side chain contacts that are more loosely defined. There have been many efforts to characterize the geometry of these contacts [12–20]. For example, Reddy and colleagues [16,18] studied the packing between a helix and a strand in 5,975 distinct α/β units that they extracted from native proteins. They reached the conclusion that the interaxial distance between the α-helix and the β-strand is linearly correlated with a complex residue-dependent function and that this information can be used for predicting interaxial distances between helices and strands. Of particular relevance to our current study, Hu and Koehl [20] analyzed the packing geometry between a helix and two strands in an $\alpha\beta_2$ motif, where the helix is in contact with two adjacent strands from one β-sheet. They collected 31,949 such $\alpha\beta_2$ motifs from a database of 6729 non-redundant high resolution protein structures, and classified them into four groups according to different orders of helices and strands in the sequence. They designed three geometric measures (see section 2.1 below) that describe the packing pattern between a helix and a sheet, and showed that the distributions observed for each measure are not random in native protein structures. From these distributions they were able to derive statistical potentials that evaluate how native-like the packing is between helices and β-sheets in a protein model; these potentials were encouragingly successful on well-known decoy data sets [20].

In this paper, we develop an automatic tool, PackHelix, that complements BuildBeta for generating protein structure models that contain both helices and strands. PackHelix samples the conformational space of helix positions relative to β-sheets using the knowledge of helix-sheet packing patterns observed in native proteins [20] to perform a controllable sampling. The goal of this sampling is to build a collection of initial configurations for later rotamer replacement and energy minimization, and with enough samples, we hope one of them will lead to a local energy minimum near the native structure. The paper is organized as follows. In the next section, we describe PackHelix and its integration with BuildBeta. In the results section, we apply BuildBeta and PackHelix to three different sets of $\alpha\beta_2$ sandwich proteins. Finally, we conclude the paper with a discussion of our results.

2 Methods

2.1 Geometric measures that characterize helix-sheet packings

When a helix is to be placed adjacent to a β-sheet, its relative position can vary due to the many possible interactions of side chains from the helix and the strands, taking into account the limited flexibility provided by the loop connecting the helix and strand. Early researchers observed some geometrical patterns for helix-sheet packing from a limited set of native proteins. Chothia *et al.* [14] studied the packing between an α-helix and a pleated β-sheet. They concluded that the helix tends to be parallel to the sheet and on the side of the sheet that makes the $\beta - \alpha - \beta$ backbone path curl as the thread of a right-handed screw. A few studies further explored helix-sheet packing [17–19] and they all agreed that the most favorable arrangement has the axis of the helix oriented along the strands of the sheet, as a result of attractive sidechain - sidechain interactions. For a parallel β-sheet, the helix axis aligns anti-parallel to the axis of the strands in the sheet [19]. Reddy *et al*. [18] showed that the mean of the distance distribution between the helix and the sheet is 10.5Å. Another energetically favorable packing, albeit less frequently observed, has the helix axis perpendicular to the strands [17, 21].

This general geometric knowledge of the helix-sheet packing is not enough to predict with reasonable accuracy the position of the helix relative to the sheet. If we specify only that a helix has its axis aligned parallel or perpendicular to the strands in sheet, it can take many possible positions. For a protein with several helices this leads to generating millions of models using a thorough sampling. From the observation of native proteins, usually a helix

does not align fully parallel or perpendicular to the strands in the sheet. In this study, we use six rigid body degrees of freedom as "packing parameters" to adjust the position of a helix relative to the sheet. They are three translation variables (D, L_1 and L_2) and three rotation angles (θ, ψ and φ) for the helix in the coordinate system based on the sheet plane, as shown in Figure 1. Once we know the values for these six variables, the position of a helix relative to a sheet is fixed. Our previous study [20] quantified the helix-sheet packing in native structures using histogram distributions of the two angles θ and ψ and of the one translation parameter D. Once these three variables are sampled, we use other information to adjust the other three degrees of freedom, as follows. Because the loops connecting the helix and strand can be short or long, the two translation variables L_1 and L_2 parallel to the sheet are adjusted to move the helix into a position where the loops are long enough to connect it to its neighboring strands. In addition, the side chains in the interaction between a helix and an adjacent sheet usually form a hydrophobic core. So we rotate the helix along its axis, by changing the φ variable, to make its most hydrophilic residues face out from the sheet and its most hydrophobic residues face towards to the sheet. Given a helix to be packed to a target sheet, we first sample the distributions of the three variables θ, ψ and D, and for each sample, we adjust the other three variables L_1, L_2 and φ according to the properties of its connecting loops and hydrophobic/hydrophilic residues. In this way, we generate several models sampling all probable positions for the helix, and some of them may be reasonably close to the position of the helix in the native protein.

2.2 Sampling in an $\alpha\beta_2$ local motif

Our PackHelix procedure works as the next module after the BuildBeta module in the ProteinShop platform. Starting with a secondary structure prediction file, BuildBeta automatically generates possible β-sheets from the sequence, with an initial helix packing to place the helices parallel to the β-sheet. PackHelix further packs the helices to the β-sheet. Instead of considering the whole β-sheet, which is usually curled and twisted, as the sheet plane to pack the helices, two paired strands are used to calculate a local sheet plane. So the helix is packed based on the information from an $\alpha\beta_2$ local motif. This is natural when the chain sequence has the helix between two strands that are neighbors in the sheet. When these two strands are not neighbors, the two loops connecting the helix and strand are checked. We select the strand connected to the helix with the shorter loop as the "base" strand, and we pick the other strand as the one of the two sheet neighbor strands of the base strand that is closer to the helix in the chain sequence. Thus, a local $\alpha\beta_2$ motif is chosen from the backbone chain and the position of helix is initially determined by the local β-motif. According to our previous analysis of helix-sheet packing [20], we divide these $\alpha\beta_2$ motifs into four groups, P0T0, P0T1, P1T0 and P1T1, because the helix-sheet packing pattern in these four groups are different. P0 means the two strands in the $\alpha\beta_2$ motif are anti-parallel while P1 means the two strands are parallel. T1 describes the situation that two strands are consecutive strands in sequence order and the helix is between them, while all other sequence orders for two strands and one helix belong to the T0 group. (For more details, see [20].)

In the analysis presented in [20] we used the three variables θ, ψ and D to describe the geometric pattern for helix-sheet packing, and we calculated the distributions of the three variables for each group. As the first step in sampling in an $\alpha\beta_2$ local motif, PackHe-lix enables the user to set sampling values on the distribution plot for each of these three variables, or to automatically generate default sampling values. θ, ψ and D are set independently of each other, as we did not detect any significant correlations between these variables in the database described in [20] (result not shown). The automatic sample generation takes the mode of the distribution as the first sampling value and distributes the rest of sampling values around the mode as shown in Figure 2. Using this user-controlled

sampling method, we can generate a reasonable number of sample values to cover the whole configuration space, so that most helix-sheet configurations in native structures have their geometric measurements close enough to one of our samples. Even if a helix-sheet packing in the native protein structure has one variable in the middle between two of the samples for one of the three variables, we hope that the subsequent energy minimization can slightly push the helix in one of our samples close to its position in the native protein.

The sampling for the variables θ, ψ and D is defined by the number of samples for each variable (given in the form of a triplet (n_t, n_p, n_D)) and the sampling values themselves.

2.3 The impact of loop constraints

A "loop" or "coil" is the part of a protein chain that connects two secondary structures, α-helices or β-strands. In an $\alpha\beta_2$ local motif, the loops connecting the α-helix and the nearby β-strands can affect the range of possible positions of the helix relative to the local β-sheet. Intuitively, a loop with more residues can allow the helix to move farther from the local β-sheet. In ProteinShop, the inverse kinematics (IK) algorithm is deployed to move β-strands and α-helices as rigid bodies by rotating the dihedral angles in the loop regions. The same implementation of the IK method used in ProteinShop [22] is applied to deforming loops in PackHelix. Every rotatable covalent bond in the backbone of a protein can be interpreted as a joint with a single axis of unconstrained rotation. After PackHelix decides on a position for a helix, this helix is selected to be moved to its target position. Loops located on both sides of this helix are activated for the IK process, so that their flexibility will allow the helix to move without altering the positions of the fixed β-strands or helices that have already been placed into their desired positions. There are two rotational degrees of freedom from the φ and ψ dihedral angles at each residue in the activated loops, and, given the position of one of the fixed structures to which the loops connect, the position of the moving helix can be predicted from these degrees of freedom using a product of rotation matrices. The Jacobian matrix of derivatives of this moving position can also be computed, by replacing in the product the rotation matrix corresponding to each degree of freedom by its derivative. We then apply the transposed Jacobian method of Welman [23] to rotate each joint by a small incremental amount. Intuitively, this method rotates each joint by an amount proportional to the torque from a force attempting to move the helix from its current position to the desired one, assuming all other joints are rigid. Because the rotation matrices involved are non-linear functions of the φ and ψ dihedral angles, we iterate in small steps, and recompute these matrices at each step. The maximum stretching length for a loop in IK can be pre-determined. We tested the maximum length L of a loop, for different numbers N of residues, to generate an approximate relation between L and N. We noticed that the shape of a loop along its path connecting a helix and a strand can also affect the maximum length it can stretch. To quantify the path shape, we count the number of turnings (defined below) along the path. In addition, because a proline residue has only one rotation degree of freedom while other residues have two, when a proline is located at either end of the loop, its presence will decrease the maximum stretching length. Thus, in our stretching length test, we classify the different cases of loops, with integer descriptors U and P, defined as follows:

- U defines the number of turnings along the path, and it can be 0, 1 or 2. We consider two secondary structures S_1 and S_2 in sequence order and a loop between them. The two secondary structures, strand or helix, can be considered as sticks and their vectors are denoted by V_1 and V_2. The vector connecting the S_1 end of the loop to the S_2 end is denoted by C. If $V_1 \cdot C$ is negative, we say there is a turning between S_1 and the loop. Similarly, if $C \cdot V_2$ is negative, a turning occurs between the loop and S_2.

- P defines the number of proline residues at the ends of a loop. Its value can be 0, 1 or 2, if there is no proline residue, one proline residues or two proline residues at the ends of the loop, respectively.

In our test, we pre-made three general cases for 0U, 1U and 2U in Figure 3. For each U and P case, we dragged the helix by the inverse kinematics in the specified direction shown by the black arrow until it could not be stretched any more, and we recorded the stretching length L in Table 1. In this table N starts from 4, because the inverse kinematics needs at least 4 residues to compute the loop between two fixed ends. This does not mean the loops that have less than 4 residues have zero stretching length. In practice, some conservative length values can be assigned to the loops with less than 4 residues.

After the initial sampling for an $\alpha\beta_2$ local motif determines the θ, ψ and D variables, the relative position of a helix with respect to the local β-sheet is still not fixed. The two translation degrees of freedom L_1 and L_2 that defines the position of the center of the helix in the plane of the β-sheet (see Figure 1 for details) are still unknown. These two degrees of freedom are in fact controlled by constraints on the length of the two loops that connect the helix to the strands. Let us consider for example the loop C_1, whose end points E_a and E_b are defined as the positions of the C_α atoms in the residues located at the end of the loop (see panel A in figure 4 for details). The coordinates (x_{E_a}, y_{E_a}, z_{E_a}) of E_a in the (s_1, s_2, n) coordinate frame associated with the β-sheet can be expressed as a function of the helix length L_h, the coordinates (L_1,L_2,D) of the helix center C_h, and the azimuthal and elevation angles θ and ψ that defines the orientation of the helix axis:

$$\begin{aligned} x_{E_a} &= L_1 + \tfrac{L_h}{2}\cos\psi\cos\theta \\ y_{E_a} &= L_2 + \tfrac{L_h}{2}\cos\psi\sin\theta \\ z_{E_a} &= D + \tfrac{L_h}{2}\sin\psi. \end{aligned}$$

The distance between E_a and E_b cannot be larger than L_{C_1}, the maximal stretching length of the loop C_1 (see above). This leads to the constraint that the projection of the helix center C_h must lie inside the circle in the sheet plane whose equation is:

$$\left(L_1 + \frac{L_h}{2}\cos\psi\sin\theta - x_{E_b}\right)^2 + \left(L_2 + \frac{L_h}{2}\cos\psi\cos\theta - y_{E_b}\right)^2 = {L_{C_1}}^2 - \left(D + \frac{L_h}{2}sin\psi - z_{E_b}\right)^2 \quad (1)$$

The center of this circle C_{R1} is ($x_{E_b} - \frac{L_h}{2}cos\psi sin\theta, y_{E_b} - \frac{L_h}{2}cos\psi cos\theta$) and its radius r_1 is $\sqrt{{L_{C_1}}^2 - \left(D + \frac{L_h}{2}sin\psi - z_{E_b}\right)^2}$. There is a similar constraint for the loop C_2 at the other end of the helix that leads to a second circle (see panel B in Figure 4). If the two circles corresponding to the two loops do not overlap, it is impossible to meet the two loop constraints simultaneously with the given θ, ψ and D and this sample can be removed from the set. If the two circles overlap, we choose the values of L_1 and L_2 as follows. If the center C_s of the reference frame of the local sheet belongs to the overlap region, we set L_1 and L_2 to be zero so that the projection of helix center C_h on the sheet plane matches with C_s. Otherwise, we define the point C_h' as the intersection of the line joining the centers C_{R_1} and C_{R_2} with its perpendicular line that joins the two points of intersection of the two circles, and we adjust L_1 and L_2 such that the projection of C_h on the sheet plane matches C_h'.

2.4 Forming a hydrophobic core

The hydrophobic effect is the dominant force in protein folding; it defines how the amino acids organize in space, with the hydrophobic amino residues mostly pointing towards the

core of the protein and the hydrophilic residues pointing towards the solvent that surrounds the protein. As such, hydrophobicity influences the relative position of the secondary structures of a protein; this effect was quantified for example with the hydrophobic dipole moment introduced by Eisenberg and colleagues [24–26]. We have used the same concept to refine the orientation of an helix with respect to its two adjacent strands when building a $\alpha\beta_2$ motif: the rotation degree of freedom φ along the axis of the helix can be determined to make its hydrophilic side chains face out from the sheet. For each helix, a hydrophilic vector is computed based on its hydrophilic side chains using the following method, shown in Figure 5. First, the centroid C_{ri} of the side chain for each hydrophilic residue i is calculated, and the vector $V = C_{ri} - C_{\alpha i}$ from its $C_{\alpha i}$ atom to C_{ri} is computed. This vector is projected onto a plane perpendicular to the axis of the helix, and normalized in the unit circle whose center is the projection h' of the axis onto that plane, to give the vector V_i for residue i. We sum up such normalized vectors from all the hydrophilic residues in one helix and the normalized sum becomes the hydrophilic vector V_h for this helix: $V_h = \frac{\sum_{i=1} V_i}{|\sum_{i=1} V_i|}$. Then we rotate the helix around its axis with the angle φ to make the minimum possible angle between its hydrophilic vector V_h and the normal n of the local β sheet that the helix is attached to. After such a rotation, most of the hydrophilic side chains will be placed on the surface of protein, while most of the hydrophobic side chains will be between the helices and the sheet, as shown in Figure 6.

2.5 Sheet side decision

After the φ variable is fixed for a (θ, ψ, D) sample by using the hydrophobic core feature, all six degrees of freedom are determined for computing the position of a helix relative to one side of the local sheet. However, it is still not certain which side of the sheet the helix should be placed on. For two parallel β strands in the $\alpha\beta_2$ motif, the right hand rule [13, 14] can be applied to predict the sheet side. In a sample where a helix is attached to a local sheet of two anti-parallel strands, it could be placed on either side. Thus we split this sample into two samples with the same six variables but using different sides of the sheet.

2.6 Multiple helices between two strands

In native protein structures, it is common that two strands that form a β-sheet are separated by more than one helix along the sequence, with all these helices packed on the sheet. In our previous analysis of the $\alpha\beta_2$ local motif, we considered each helix separately in such a case. However, when multiple helices are attached to the same two strands, they cannot each be simply packed individually without considering the others. From the observation of native protein structures, we propose the following rules for packing multiple helices between two strands based on helix size. A helix is considered to be small if it has less than 8 residues, of medium size if it has between 8 and 20 residues, and large otherwise. Similarly, a loop with less than 4 residues is said to be small.

- If a small helix is adjacent to one of the two strands in the $\alpha\beta_2$ motif, with a small loop connecting it to that strand, it is considered as part of the (longer) loop connecting the strand to the next helix in the sequence. It is packed perpendicular to the sheet and beyond the end of its neighboring strand. For such helices, the variable φ is set to 90 degrees.
- If a large helix is found among four or more helices between two strands, it will be placed further away from the sheet than the smaller helices. For such helices, the D variable is set to twice its normal value.

Besides the two rules above, an adjustment method is used to avoid collisions between the multiple helices. We have implemented a coarse-grained method with the two helices represented as sticks, rather than an exact method that works at the atomic level. We believe

that this is appropriate as the positions of the side-chains are going to change in subsequent rotamer replacement and energy minimization stages. We detect the collision between two helices by simply calculating the minimum distance between their two line segments. For each helix, a collision detection test is applied with respect to each helix that has been already packed. If one of the corresponding minimum distances is found to be smaller than a threshold set by the user, the center of this helix is moved away from the helix it intersects, as explained in Figure 7 and its caption. It should be noted that this helix adjustment method may fail because of the loop length constraints. The program attempts to move the helix away from the other helices it intersects to the extent that its connecting loops allow.

2.7 Helix packing with a tree structure

The helix packing procedure goes through all helices in sequence order. Each helix can have a number of different positions equal to the number of initial sample sets of θ, ψ and D choices times its possible sheet sides, so the number of total structures with different helix-sheet packing is exponential in the number of helices. For example, suppose a sequence has 5 helices and each one is assigned with three initial samples on only one sheet side. Then there are at most 3^5 tertiary structures with different helix-sheet packing onto the same sheet. A tree data structure is introduced for the algorithm to generate all structures with all possible samplings of the helix-sheet packing onto a single sheet, with the path from the root to a node at level n having a particular choice of the (θ, ψ and D) sample set for the first $n - 1$ helices. The depth of the tree is the number of helices, and each level is in charge of one particular helix. A tree node has a list of entries describing its children. Each entry for a node at level n represents one set of sample for values of θ, ψ and D for the n^{th} helix. A depth first traversal (DFT) can be applied to visit all entries in this tree. Each time it reaches an entry in a leaf node, the path from the root to the leaf determines entries at each intermediate node, which are used to generate the relative positions for all helices to the their attached sheet. There are two advantages in using this tree in the model generation of packed helices.

- The status of the structure undergoing helix packing is checked at every node of the tree, and once one of its error scores, such as a collision or an IK convergence error from insufficient loop flexibility, exceeds the threshold value set by the user, the subtree of its node can be killed without traveling into it. The collision score between any two atoms is set to be three fourths of the sum of their covalent radii minus the distance between their centers with a floor of zero (i.e. it is set to zero if this quantity is negative). The total collision score for a model is the simply sum of the collision scores between each pair of atoms in the model.
- Because the IK computation is the most expensive stage of the helix packing, the positions of atoms for the whole structure are saved in the head of every node in a temporary storage space. Every time an entry that is not the first entry in a node is reached, the structure is reset with the saved atom positions, before using IK to pack the new helix according to the packing parameters in this entry. It is much faster to reset atom positions for the whole model than to move all the previous helices to their appropriate original positions by using IK. Since DFT is used here, the saved position array for a node is deleted after its last entry is visited. The maximum number of sets of atom positions at one time is the depth of the tree, which is the number of helices.

2.8 A pipeline to generate models for *αβ*-proteins

The general pipeline we propose to generate model structures for $\alpha\beta$-proteins consists of four steps: (i) definition of the number, type and sequence location of the secondary structure elements of the protein considered, (ii) generation of possible topologies and

geometries for the β-sheets identified in the proteins using BuildBeta, (iii) packing of the helices on the β-sheets using PackHelix, and (iv) rotamer replacement for the side-chains followed by an energy minimization.

In a real application of this pipeline, the secondary structures of the protein of interest are not known. We therefore use the results of secondary structure prediction servers, such as PSIPRED [27] and JUFO [28]. In test cases with known native proteins, we use the native assignment of secondary structures.

BuildBeta, given enough space and time, creates all possible arrangements into β-sheets of the β-strands identified in the protein. As the number of these arrangements can be large, BuildBeta orders the possible strand topologies by the scores of Ruczinski [10] so that the most probable can be considered first. The number of arrangements passed to the PackHelix step is controlled by the user.

PackHelix is subsequently run on the models generated by BuildBeta to pack the helices identified in the protein. PackHelix uses pre-set sampling numbers for the geometric measures θ, ψ, and D that defines the position of a helix with respect to the β-sheet it is attached to (see above). Usually larger sampling numbers lead to a larger number of result models. The number N of models that PackHelix generates is a function of the sampling numbers (n_θ, n_ψ, n_D), of the number of helices (n_h), of the number of helices that can be placed on both sides of the sheet (n_{bs}), and of the number of sheet models (n_m) for the input:

$$N = n_m (n_\theta n_\psi n_D)^{n_h} 2^{n_{bs}}.$$

Note that many of the models are likely to be eliminated due to the large number of collisions that cannot be fixed, so the final number of models PackHelix generates is usually much smaller than N. Ultimately, this number is controlled by the user.

The models generated by PackHelix are subsequently ranked, using the combination of scoring functions listed above. The sidechains of all residues in the corresponding top scoring models are then repacked using the program SCWRL4 [29]. Finally, these models are subjected to energy minimization to improve their stereochemistry and to remove any potential steric clashes, using GROMACS [30].

3 Results

The current implementation of PackHelix is limited to proteins that contain $\alpha\beta_2$ motifs, where the helix is in contact with two strands from one β-sheet. As described in the methods section above, complications occur when several helices that appear between two consecutive strands in the sequence of the protein interact with the sheet that includes these two strands. We have therefore considered two test sets for PackHelix. Test set 1 includes several medium-sized proteins, for which at most one helix is found between two consecutive strands in the sequence, while test set 2 includes proteins for which multiple helices are found between two consecutive strands. We have removed in these proteins any long coil regions at their N and C termini to focus on their compact $\alpha\beta$ core. The two test sets 1 and 2 were designed to assess the ability of PackHelix to pack helices correctly on a known β-sheet. As such, they contain proteins whose native structure is known. In addition, we have used their native secondary structure assignments as well as the correct topologies for their β-sheets as input to PackHelix. Namely, we picked one of the best models generated by BuildBeta which has the same sheet topology and similar alignments between adjacent strands as those of the native structure. Finally, the best models reported in table 2

are the structures with the smallest RMSD to their corresponding native structures; these RMSD refer to the coordinate RMS between C_α atoms.

We also include as a blind test one actual prediction case from CASP9 as test set 3, to test PackHelix on a real application. For this case, we selected a few models from the results generated by BuildBeta based on predicted secondary structures and we ran PackHelix on each of these models. The five models submitted to CASP9 were selected using a combination of two knowledge-based scoring functions: Dfire [31] and RAMP [32]. Dfire [31] is an efficient distance-based atomic statistical potential with 167 atom types (i.e. all heavy atom types in the 20 types of amino acid) that uses a ideal gas reference state. RAMP [32] is a suite of programs for analyzing protein structures developed by Samudrala and colleagues. It includes two distance-based statistic potentials, based on local and non-local information, respectively; we have used the latter. The ”best” model reported here is the closest of these five models to the native structure (which was released after CASP9 ended), based on RMSD.

The computing time required by PackHelix to pack helices on their adjacent β-strands is directly related to the number of values n_t, n_p, and n_D chosen to sample θ, ψ, and D, respectively. To keep the computing time within reasonable values, we have chosen $n_t = 2$ or 3, $n_p = 2$ or 3, and $n_D = 1$ in all test cases described below; the corresponding values for θ, ψ and D are given in table 2.

3.1 Test set 1

This group includes 3-layer $\alpha\beta\alpha$ sandwich proteins between 100 and 200 residues. The number of α-helices in each protein is between 4 and 6. There are no multiple helices between two consecutive strands in sequence order, and most of the helices are of medium length (i.e. between 8 and 20 residue long). Specifically, the set includes: the sporulation protein N-Spo0A from bacillus stearothermophilus (PDB code 1DZ3, chain A, from residue 1 to residue 114), a phosphopantetheine adenylyltransferase from E. coli (PDB code 1B6T, chain A, from residue 3 to residue 121), the PhoB receiver domain from E. coli (PDB code 1B00, chain A, from residue 8 to residue 123), and a rat NADPH-cytochrome P450 reductase (PDB code 1AMO, chain A, from residue 64 to residue 235).

1DZ3 A—This protein has a flavodoxin-like fold; it has an alternating α/β topology arranged such that four helices are packed on a sheet of five parallel strands, with 3 helices on one side and the fourth one on the other side. We removed the C-terminal helix in the structure available in the PDB file 1DZ3 as this helix interacts with chain B. The best model generated by PackHelix has a 3.82 Å RMSD to the native structure; it is shown in Figure 8 (gray color) superimposed on the native structure (in purple and magenta colors). The sampling numbers were set to (3,2,1) for (θ, ψ, D) and PackHelix generated a total of 1621 models. Figure 8 shows a very good alignment between the packed helices in our best model and those in the native model, which indicates that our sampling method covers a dense enough sampling set of arrangements of the helices to include a structure close to the native one.

1B6T A—Similar to 1DZ3 A, this target has an alternating α/β topology arranged such that four helices are packed on a sheet of five parallel strands, but with two helices on one side and two helices on the other side. The C-terminal helices in the structure available in the PDB file were not considered as they do not interact with the β-sheet. Our best model has a 4.11 Å RMSD to the native structure and it is shown in Figure 8 superimposed on the native structure. The sampling numbers were set to (3,3,1) for (θ, ψ, D) and PackHelix generated 3945 models. This figure again shows that with a dense sampling set, PackHe-lix can

generate models whose helix positions are very close to those found in the native protein structure.

1B00 A—This protein has a flavodoxin-like topology, with five helices packed on a sheet of five parallel strands. There are two helices on one side of the sheet and three helices on the other side. Our best model has 5.36 Å RMSD to the native structure and it is shown in Figure 8, superimposed on the native structure. The sampling numbers were set to (3,3,1) for (θ, ψ, D) and PackHelix generated 17015 models. The best model generated for this protein is not as good as those generated for 1DZ3 A and 1B6T A (in terms of RMSD). This comes from the fact that the native structure has one long loop (8 residues) between strand 2 and helix 3 and our inverse kinematics procedure does not generate a native-like shape for long loops.

1AMO A—The C-terminal domain of the rat NADPH-cytochrome P450 reductase (PDB file 1AMO, A chain from residue 64 to residue 235) has a ferredoxin-reductase-like fold, with five helices packed on a sheet of five parallel strands. There are two helices on one side of the sheet and three helices on the other side. The sampling numbers were set to (2,2,1) for (θ, ψ, D) and PackHelix generated 875 models. Our best model shown in Figure 8 has a 6.37 Å RMSD difference from the native protein; this large discrepancy is due to a long loop at the C-terminal end, whose shape cannot be modeled properly by inverse kinematics.

3.2 Test set 2

This group also includes 3-layer $\alpha\beta\alpha$ sandwich proteins between 100 and 200 residues long. The number of α-helices in each protein is between 5 and 9. In contrast to test set 1, there are multiple helices between two consecutive strands in sequence order in these proteins. The lengths of the helices can be very different, from 3 residues to 25 residues. Specifically, the set includes: 3-dehydroquinate dehydratase from mycobacterium tuberculosis (PDB code 1H05, chain A, from residue 3 to residue 137), the enzyme glycerol-3P cytidyltransferase from bacillus subtilis (PDB code 1COZ, chain A, from residue 1 to residue 113), the type II dehydroquinase from streptomyces coelicolor (PDB code 1D0I, chain A, from residue 7 to residue 150), a response regulator for cyanobacterial phy-tochrome (PDB code 1I3C, chain A, from residue 9 to residues 138), and the ycaC gene product from E. coli (PDB code 1YAC, chain A, from residue 12 to residue 175).

1H05 A—This protein has a flavodoxin-like fold, with eight helices packed on a sheet of five parallel strands. Three pairs of strands in the sheet have two helices between them, a small helix functioning as a loop and a longer helix interacting with the sheet. We have set the sampling numbers to (2,2,1) for (θ, ψ, D) and PackHelix generated 626 models. The best model PackHelix generated has a 4.68 Å RMSD difference with the native structure. It is shown in Figure 9, with all small helices packed in positions that are very close to their positions in the native protein structure; this supports our rule for packing small helices when multiple helices are between two consecutive strands.

1COZ A—This protein has an adenine-nucleotide-alpha-hydrolase-like fold, with six helices packed on a sheet of five parallel strands. There is a small helix (three residues long, from residue 100 to residue 102) that is directly connected to a strand, without a loop between them. As the inverse kinematics procedure cannot move such a helix, it was converted into a loop. Therefore there are five helices in our model with two of them between two consecutive strands. The sampling numbers were set to (2,2,1) for (θ, ψ, D) and PackHelix generates 741 models. Our best model shown in Figure 10 (top left) has a 5.08 Å RMSD difference with the native structure, with all the helix positions very close to their positions in the native structure. When this is compared to the best model (6.3Å RMSD)

generated using the simple helix packing method implemented in BuildBeta [8], we see that PackHelix improved this model by providing superior helix packing to the same sheet.

1D0I A—This protein has a flavodoxin-like fold, with nine helices packed on a sheet of five parallel strands. Three pairs of strands in the sheet have multiple helices between them, up to three helices (helices at position 16–20, 24–29 and 32–46 between strand 1, 9–13 and strand 2, 52–55). It is very challenging to pack 10 helices appropriately on a single sheet with five small strands because of steric constraints. We have set the sampling numbers to (2,2,1) for (θ, ψ, D) and PackHelix generated 559 models. Our best model (Figure 10) shows that our method for packing multiple helices between two consecutive strands is very successful for this protein. Our best model has a 5.8 Å RMSD with respect to the native structure, with a native-like pattern for the multiple helices between two consecutive strands.

1I3C A—This protein has a flavodoxin-like fold, with six helices packed on a sheet of 5 parallel strands. One pair of strands in the sheet has two small helices between them, and one small helix is unusually far away from the sheet. The sampling numbers were set as 2,2,1 for (θ, ψ, D) and PackHelix generated 468 models. None of the results from PackHelix exactly pack the latter small helix to its unusual position in the native protein. Our best model, shown in Figure 10, has a 6.75 Å RMSD with respect to the native structure, primarily because our IK was not able to generate native-like loop shapes for two long loops in this protein.

1YAC A—This protein has a isochorismatase-like hydrolase fold, with seven helices packed on a sheet of six parallel strands. Two pairs of strands in the sheet have two helices between the two strands in the pair. The sampling numbers were set to (2,2,1) for (θ, ψ, D) and PackHelix generated 1667 models. Our best model shown in Figure 10 has a 6.1 Å RMSD with respect to the native structure.

3.3 Test set 3

This group consists of one target from CASP9, in which we participated. We considered this protein as an $\alpha\beta\alpha$ sandwich protein. Starting from the secondary structure prediction obtained from PSIPRED [27] and JUFO [28] we first ran BuildBeta to model its β-sheet. We then picked the five best models from the results of BuildBeta (according to the Build-Beta assessment, see methods above) and subsequently applied PackHelix to position their helices.

T0517—This target was predicted by the secondary structure prediction servers to have seven helices with a sheet of 5 strands. The most challenging part of this target is that there are three small to medium size helices and one very large size helix between two consecutive strands in sequence order. In addition, the loops between these four helices are not long enough to provide a good flexibility for packing the helices: It is therefore difficult to arrange these four helices in a pattern so that they do not cross each other while still keeping the whole protein structure compact. PackHelix was run on five different sheet arrangements from the results of BuildBeta; in each case, the sampling numbers for (θ, ψ, D) were set to (2,2,1). PackHelix only generated a total of twenty structures as most of the models were eliminated due to their large collision score. Our best model from five submitted structures, shown in Figure 11, satisfies the constraint that the structure is compact and the helices do not collide with each other.

T0517 is a putative sulfurtransferase (dsrE(Swol 2425) from Syntrophomonas wolfei str. Goettingen). Its structure has been determined experimentally by X-ray crystallography and has recently been been released (PDB code 3PNX, 159 residues): it is shown on the left

panel in Figure 11. The RMSD (C_α atoms only) between the native structure and the model we deemed best at the time of submission to CASP (shown as the right image in Figure 11) is 13.6 Å. We note that this poor performance is mostly due to imprecise secondary structure prediction and not to PackHelix *per se*. The native structure shows that this protein contains a β-sheet of four strands, while the secondary structure prediction servers predicted a sheet with five strands. The last strand that is close to the C terminus in our model is actually a helix in the native structure. This incorrect prediction made it difficult for BuildBeta to generate correct β-sheets topologies; we do notice that it did relatively well on the other four strands. The long helix in our model should in fact be two helices with a middle sized loop between them. This wrong prediction led PackHelix to generate the wrong pattern for packing this helix. This blind test result emphasizes that although our PackHelix can create native-like helix packing even for models that have very dense helices, the result is greatly influenced by any error in the initial secondary structure prediction.

3.4 Running time requirements

The most time-consuming part of running PackHelix is applying inverse kinematics to generate the loops when PackHelix moves helices to their target positions. As shown in Table 2, it may takes up to several days to run PackHelix on a desktop computer if we set relatively large sampling numbers for a protein that contains multiple helices.

4 Discussion

ProteinShop [9] is a stand-alone software platform designed to predict, manipulate, and visualize protein structures. It is the sum of many parts that cover different aspects of protein structure, including knowledge from our understanding of the physics and chemistry that define the stability and geometry of a protein, as well as statistical knowledge extracting from the databases of known protein structures. It is in constant evolution, introducing new tools designed to expand its field of applications as well as improve the quality of the predictions it generate. Recently, we have introduced BuildBeta [8], a tool to predict the topology and geometry of β-sheets in proteins. In this paper, we describe PackHelix, a post-processing tool for $\alpha\beta$ proteins, used after BuildBeta, to optimize the position of helices that pack on the β-sheet generated by BuildBeta. BuildBeta itself has a simple helix packing strategy, that basically positions a helix that interacts with a β-sheet parallel to the sheet with its axis in the opposite direction compared to the direction of the strands. It works well for the case of a single helix packed between two parallel strands, which is a very common $\alpha\beta\alpha$ topology in sandwich-style proteins. It is likely to fail, however, for proteins that have multiple helices between two consecutive strands in the sequence or for positioning a helix between two strands that are anti-parallel to each other. Even for the generic $\alpha\beta\alpha$ motif, it is limited to positioning the helix parallel to the strands, while surveys of helix strand packings have revealed that many orientations of the helix are possible [20].

To the best of our knowledge, there are currently very few alternative methods to Build-Beta that focus on the problem of packing helices and β-sheets. The most significant exception is the work of Taylor and colleagues [4, 5], that was inspired by earlier work by Finkelstein and co-workers [33,34]. Basically, they assume that proteins are built around a library of ideal motifs or forms, consisting of layers of α-helices and β-sheets. These forms are represented as “boxes” connected by sticks that are constrained to be compact. While this approach was shown to build native-like models successfully for a set of proteins of less then 200 residues [5], we do see possible limits for its generalization. Mainly, it is strongly anchored in the concept that the number of folding units are limited (the so-called “periodic table” for protein structures [4]) and that these units can be built using ideal geometries and topologies. Recent analyses of protein structures have shown large variations in the packing geometries of proteins (see for example [17, 20]).

PackHelix presented here is an automatic tool for packing helices on β-sheets for $\alpha\beta\alpha$ sandwich proteins, that was designed to solve the inherent problems listed above. It utilizes the knowledge of packing form deduced from native $\alpha\beta\alpha$ sandwich proteins and uses the statistical distribution of rigid body motion parameters to sample wisely with user controls. Therefore, the sampling set generated by PackHelix contains different packing angles between the axes of a helix and a strand. The number of samples for the variables (θ, φ and D) and their sampling values can be customized by the user. In the test cases presented here, we have used widely spaced sample values to cover as much as possible the full conformational space for the helix position. PackHelix can also pack multiple helices reasonably well. In addition, PackHelix has features such as the loop length constraint, building a hydrophobic core, and the collision-minimization helix adjustment, to ensure that its resulting models have very few collisions. Our results show that PackHelix improves the helix packing part of the original BuildBeta, especially for models that have multiple helices between two consecutive strands, and it is able to automatically create a dense enough sampling set of arrangements for the helices to include a structure very close to the native one.

Both PackHelix and BuildBeta are solidly anchored in geometry, making use of secondary structure arrangements found in existing protein structures to predict the overall topology and geometry of a complete protein. This is similar to the principles behind the ideal motifs developed by Taylor at al [4, 5] with the significant difference that the arrangements we consider are derived from known protein structures and not from ideal geometries. In our approach, the sequence of the protein is only used to predict the secondary structure elements of the protein, to determine probable alignments of β strands in sheets using [11], and to rank the models that have been generated (by measuring the structure/sequence fitness); it is not used in the process of generating the 3D structure itself, neither by BuildBeta nor by PackHelix (with the minor exceptions of accounting in Build-Beta for proline resides in forming the hydrogen bonds that determine the relative positions of the β strands in a sheet, and of accounting in PackHelix for proline residues at the beginning or end of a loop region, see methods). This is probably inefficient as it is likely that some patterns in the amino acid sequence of a protein correlate with the parameters that define secondary structure packing and/or with the conformation of the loops that connect them. These are second order effects that are difficult to detect with statistical significance.

As observed in native protein structures, some $\alpha\beta\alpha$ sandwich proteins have very complicated packing patterns for their helices; these complex geometries are not handled properly by PackHelix (see for example the test cases 1AMO A and 1I3C A in the results section). PackHelix cannot solve all difficulties in the field of helix packing and it still needs improvements in the future. Some directions that we are considering:

- *Allow flexibility in the geometry of helices*. Currently PackHelix considers all helices to be cylinder-like (i.e. with a straight axis). This is a definite limitation, as long helices may be curved and/or twisted.
- *Improve the positioning of multiple helices attached to one sheet*. The current method implemented in PackHelix is ad hoc and requires improvement. In particular, we should look more systematically at the relative organization of multiple helices attached to the same sheet to potentially derive rules similar to those that define the packing of a single helix on a sheet.
- *Improve the generation of loops*. Inverse kinematics automatically generates loops with fixed bond lengths and angle values (i.e. the standard values from stereochemistry) but does not consider physical energy terms, such as a term for collisions. It is a key component for the automatic placement of helices on β-sheets

that satisfies a geometrically correct connectivity to the strands. We have observed however that inverse kinematics fails to generate native-like structures for long loops; this needs to be addressed to improve the performance of PackHelix.

- *Scoring the models*. Both BuildBeta and PackHelix suffers from an exponential explosion in the number of models they generate as the number of secondary structure elements increases. There is a need for scoring functions that can filter out topologies and geometries that are unlikely to be found in native proteins. As described in the method section, we have used knowledge-based energy functions as a first attempt to solve this problem. We believe however that there is much room for improvement.

The current version of PackHelix, as well as BuildBeta, will be made available as part of the upcoming release of ProteinShop [35].

Acknowledgments

C. H. and N. M. are supported by the Director, Office of Advanced Scientific Computing Research, Office of Science, of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231 through the Scientific Discovery through Advanced Computing (Sci-DAC) program's Visualization and Analytics Center for Enabling Technologies (VACET). We wish to thank Wes Bethel, head of the visualization lab at LBNL for his support. We thank also Silvia Crivelli for her useful comments on the paper. We also thank the anonymous reviewers for their valuable comments. P. K. acknowledges support from the NIH.

References

1. Moult J. A decade of CASP: progress, bottlenecks and prognosis in protein structure prediction. Curr Opin Struct Biol. 2005; 3:285–289. [PubMed: 15939584]
2. Bonneau R, Baker D. Ab initio protein structure prediction: progress and prospects. Annu Rev Biophys Biomol Struct. 2001; 30:173–189. [PubMed: 11340057]
3. Hardin C, Pogorelov TV, Luthey-Schulten Z. Ab initio protein structure prediction. Curr Opin Struct Biol. 2002; 12:176–181. [PubMed: 11959494]
4. Taylor WR. A periodic table for protein structures. Nature. 2002; 416:657–660. [PubMed: 11948354]
5. Taylor WR, Bartlett GJ, Chelliah V, Klose D, Lin K, Sheldon T, Jonassen I. Prediction of protein structure from ideal forms. Proteins Struct Funct Bioinf. 2008; 70:1610–1619.
6. Bradley P, Misura KMS, Baker D. Toward high-resolution de novo structure prediction for small proteins. Science. 2005; 309:1868–1871. [PubMed: 16166519]
7. Bradley P, Baker D. Improved beta-protein structure prediction by multilevel optimization of nonlocal strand pairings and local backbone conformation. Proteins: Struct Func Bioinf. 2006; 65:922–929.
8. Max N, Hu CC, Kreylos O, Crivelli S. Buildbeta - a system for automatically constructing beta sheets. Proteins: Struct Func Bioinf. 2002; 48:85–97.
9. Crivelli S, Kreylos O, Hamann B, Max N, Bethel W. ProteinShop: a tool for interactive protein manipulation. J Comput Aided Molec Des. 2004; 18:271–285. [PubMed: 15562991]
10. Ruczinski I, Kooperberg C, Bonneau R, Baker D. Distributions of beta sheets in proteins with application to structure prediction. Proteins: Struct Func Bioinf. 2002; 48:85–97.
11. Zhu H, Braun W. Sequence specificity, statistical potentials, and three-dimensional structure prediction with self correcting distance geometry calculation with β-sheet formation in proteins. Protein Sci. 1999; 8:326–342. [PubMed: 10048326]
12. Sternberg MJ, Thornton JM. On the conformation of proteins: the handedness of the beta-strand alpha-helix beta-strand helix. J Mol Biol. 1976; 105:367–382. [PubMed: 972389]
13. Richardson J. Handedness of crossover connections in beta sheets. Proc Nat Acad Sci (USA). 1976; 73:2619–2623. [PubMed: 183204]

14. Chothia C, Levitt M, Richardson D. Structure of proteins: packing of α-helices and pleated sheets. Proc Nat Acad Sci (USA). 1977; 74:4130–4134. [PubMed: 270659]
15. Cohen FE, Sternberg MJ, Taylor WR. Analysis and prediction of the packing of alpha-helices against a beta-sheet in the tertiary structure of globular proteins. J Mol Biol. 1982; 156:821–862. [PubMed: 7120396]
16. Reddy BVB, Blundell TL. Packing of secondary structural elements in proteins: Analysis and prediction of inter-helix distance. J Mol Biol. 1993; 233:464–479. [PubMed: 8411156]
17. Boutonnet NS, Kajava AV, Rooman MJ. Structural classification of $\alpha\beta\beta$ and $\beta\beta\alpha$ supersecondary structure units in proteins. Proteins: Struct Func Genet. 1998; 30:193–212.
18. Reddy BVB, Nagarajaram HA, Blundell TL. Analysis of interactive packing of secondary structural elements in α/β units in proteins. Protein Sci. 1999; 8:573–586. [PubMed: 10091660]
19. Hespenheide BM, Kuhn LA. Discovery of a significant, nontopological preference for antiparallel alignment of helices with parallel regions in sheets. Protein Sci. 2003; 5:1119–1125. [PubMed: 12717033]
20. Hu CC, Koehl P. Helix-sheet packing in proteins. Proteins: Struct Func Bioinf. 2010; 78:1736–1747.
21. Chou K, Nemethy G, Rumsey S, Tuttle RW, Scheraga HA. Interactions between an alpha-helix and a beta-sheet. energetics of alpha/beta packing in proteins. J Mol Biol. 1985; 186:591–609. [PubMed: 4093981]
22. Kreylos O, Max N, Hamann B, Crivelli S, Bethel W. Interactive protein manipulation. Proceedings of 14th IEEE Visualization. 2003:581–588.
23. WELMAN, C. Master's thesis. Simon Fraser University; 1993. Inverse kinematics and geometric constraints for articulated figure manipulation.
24. Eisenberg D, Weiss RM, Terwilliger TC, Wilcox W. Hydrophobic moments and protein structure. Faraday Symp Chem Soc. 1982; 17:109–120.
25. Eisenberg D, Weiss RM, Terwilliger TC. The hydrophobic moment detects periodicity in protein hydrophobicity. Proc Natl Acad Sci (USA). 1984; 81:140–144. [PubMed: 6582470]
26. Eisenberg D, Mclachain AD. Solvation energy in protein folding and binding. Nature. 1986; 319:199–203. [PubMed: 3945310]
27. McGuffin LJ, Bryson K, Jones DT. The PSIPRED protein structure prediction server. Bioinformatics. 2000; 16:404–405. [PubMed: 10869041]
28. Meiler J, Mueller M, Zeidler A, Schmaeschke F. Generation and evaluation of dimension reduced amino acid parameter representations by artificial neural networks. J Mol Model. 2001; 7:360–369.
29. Krivov GG, Shapovalov MV, Dunbrack RL Jr. Improved prediction of protein side-chain conformations with scwrl4. Proteins: Struct Func Bioinf. 2009; 77:778–795.
30. der Spoel DV, Lindahl E, Hess B, Groenhof G, Mark AE, Berendsen HJC. GROMACS: fast, flexible, and free. J Comp Chem. 2005; 26:1701–1718. [PubMed: 16211538]
31. Yang Y, Zhou Y. Ab initio folding of terminal segments with secondary structures reveals the fine difference between two closely related all-atom statistical energy functions. Protein Sci. 2008; 17:1212–1219. [PubMed: 18469178]
32. Samudrala R, Moult J. An all-atom distance-dependent conditional probability discriminatory function for protein structure prediction. JMB. 1998; 275:893–914.
33. Finkelstein AV, Ptitsyn OB. Why do globular proteins fit the limited set of folding patterns? Prog Biophys Molec Biol. 1987; 50:171–1990. [PubMed: 3332386]
34. Chothia C, Finkelstein AV. The classification and origins of protein folding patterns. Ann Rev Biochem. 1990; 59:1007–10039. [PubMed: 2197975]
35. Crivelli, S. http://sourceforge.net/projects/proteinshop

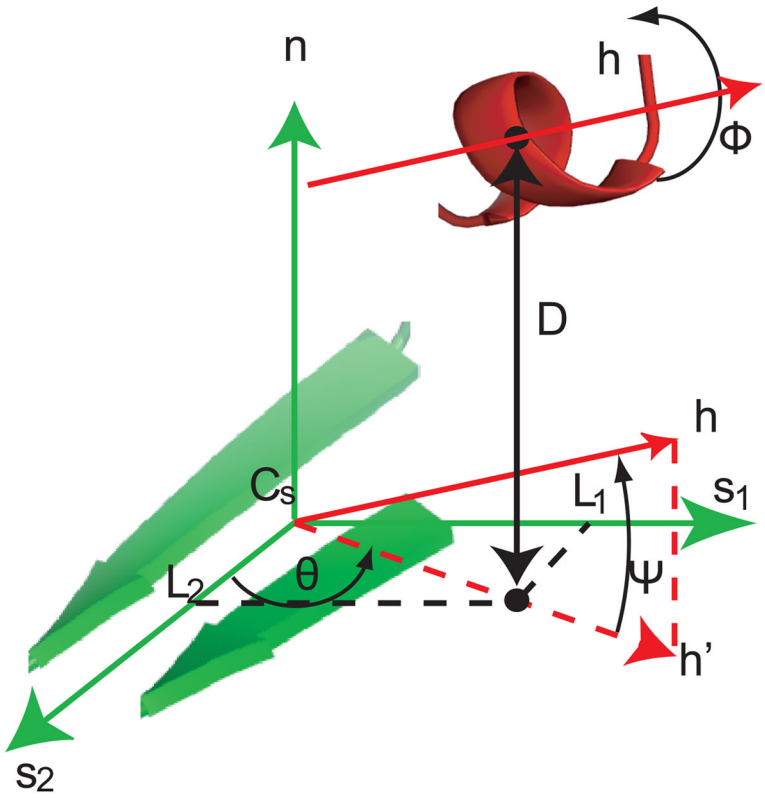

Figure 1. Six degrees of freedom for a helix in the$\alpha\beta_2$ unit

The position of the helix is defined with respect to a coordinate system anchored on the local β-sheet of two strands [20], with its origin between the two strands. s_1 is the mean vector representing the two strand axis directions, while s_2 is a vector perpendicular to s_1 in the plane with normal n containing the sheet. The origin C_s of the local sheet plane is the centroid of all C_α atoms of residues that have hydrogen bonds to each other along the two strands. The helix is considered as a stick, and the orientation of its axis h and the position of its center C_h can be defined in a polar coordinate system: the azimuthal angle θ is the rotation angle between s_1 and the projection h' of h in the plane defined by s_1 and s_2, while the elevation angle ψ is the angle between this plane and h. The angle φ is the rotation angle around the axis h itself. L_1 is the s_1 component of the translation vector for the projection C_h' of center C_h of the helix in the sheet plane away from the origin C_s, and L_2 is its component along the s_2 axis. Similarly, D is the translation component along the normal to the sheet plane.

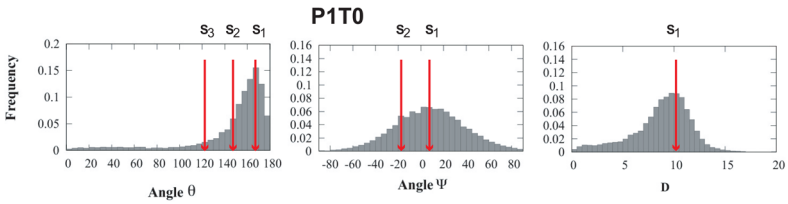

Figure 2. User-controlled sampling on three geometrical features for the P1T0 group

This figure shows the sampling sets for the θ, ψ and D variables for a helix from the P1T0 group with the numbers of samples specified by users. The number of samples for θ is set to 3 and the sampling values S_1, S_2 and S_3 are shown in the distribution graph at 165, 145, and 120. There are two samples in the distribution of angle ψ at 5 and −20. The D value is assigned with only one sample at its maximum frequency peak at 10. The first sampling value is set at the peak of the distribution while the following values are selected randomly on both sides of the peak so as to provide a good coverage of the distribution.

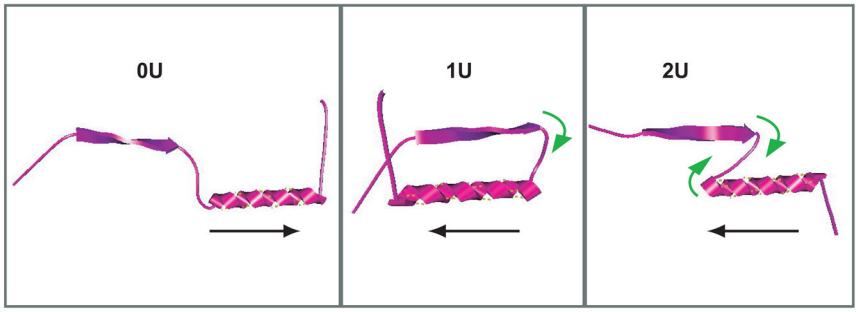

Figure 3. 0U, 1U and 2U cases in the loop stretching test

This figure shows three cases with different number of turnings in our loop stretching test. We test the maximum stretching length of a loop between one strand and one helix, with the strand being either parallel or anti-parallel to the helix, as these are the two prevalent orientations in native protein structures. In the left panel, there is no turning between the helix and the strand. The middle panel shows the 1U testing case, which has one turning between the strand and the helix. Similarly, the right panel shows the 2U testing case that has two turnings. The helix is dragged in the direction of the black arrow using IK until it can move no farther, and we record the maximum length for the loop for each case.

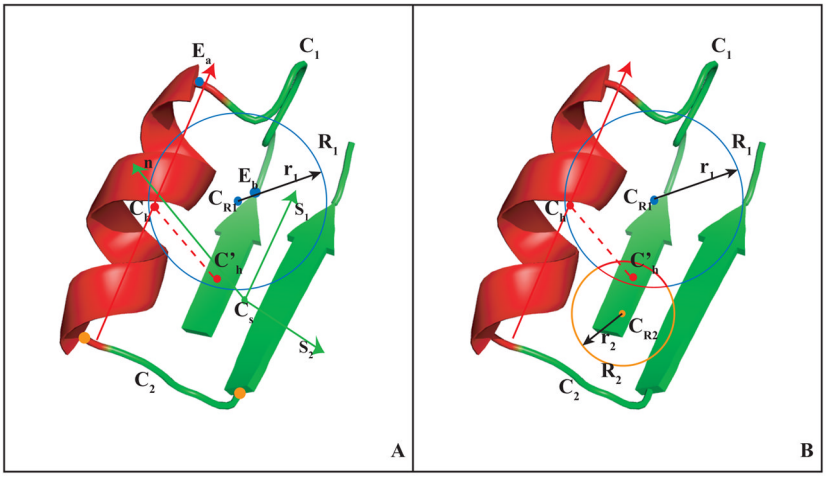

Figure 4. Using Loop constraints to position the helix with respect to theβ-sheet

Once the azimuthal and elevation angles θ and ψ and the elevation distance D are known, the relative orientation of the helix axis with respect to the β-sheet plane is known; we still need however to fix the helix itself: this is done by defining the position of $C_h^{'}$, i.e. the projection of the helix center C_h in the sheet plane. This position is constrained by the loops that connect the helix to the two strands. (A) The length of the loop C_1, with end points E_a and E_b cannot be larger than its maximum stretching length L_{C_1}; as a consequence, there are limits on much the helix can be dragged parallel to the sheet plane; these limits translate into the condition that the projection $C_h^{'}$ of the helix center lie inside a circle R_1 with center C_{R1} and radius r_1 (see text for details). (B) When both loops C_1 and C_2 are considered, the loop constraints impose that the projection $C_h^{'}$ of the helix center lie inside the overlap between the two corresponding circle R_1 and R_2.

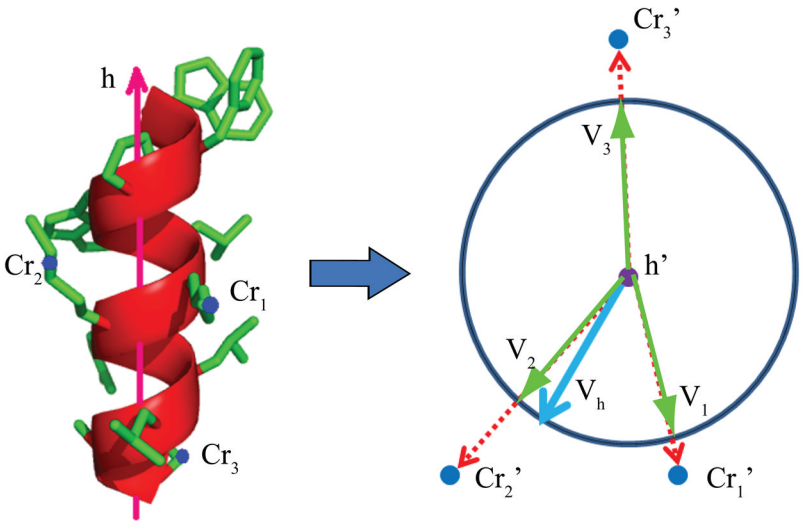

Figure 5. Computing the hydrophilic vector for a helix

The centroid of each hydrophilic residue *i* is computed (identified as a blue dot labeled Cr_i on the left panel). These blue dots are projected onto a plane $\mathcal{P}$ that is perpendicular to the axis of the helix, as shown in the right panel, to give points Cr_i'. Let h' be the intersection of the helix axis with the plane $\mathcal{P}$. The vectors $h'Cr_i'$ are normalized to unit length, yielding the vectors V_i. The hydrophilic vector V_h is the normalized sum of these vectors V_i.

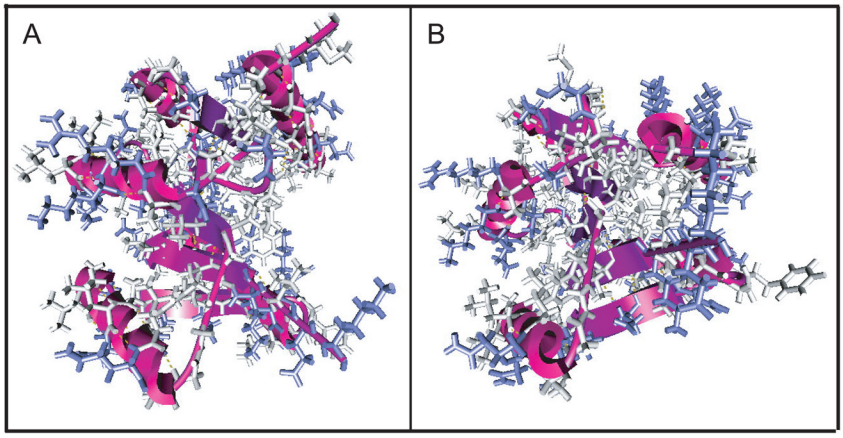

Figure 6. Improving side-chain packing by rotating the helices

We compare the models generated by PackHelix before (left panel) and after (right panel) rotating the helices to form an hydrophobic core for the bacterial chemotaxis protein CheY(PDB ID 1CHN) (see method for details). In both panels, hydrophobic side-chains are colored white, while hydrophilic side-chains are colored blue. Note the clustering of the ”white”, hydrophobic side-chains after rotation.

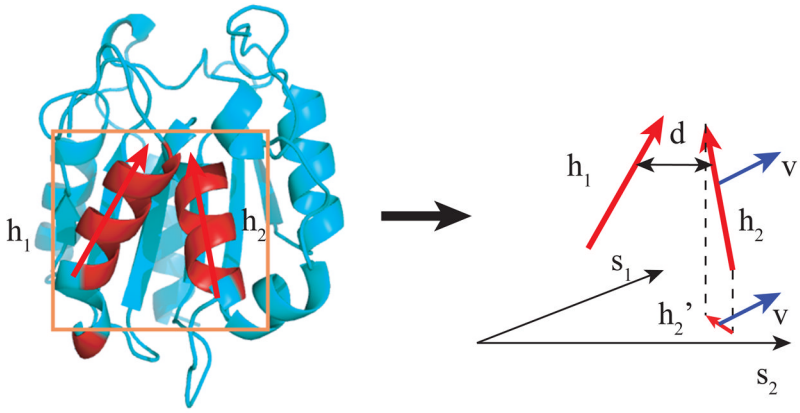

Figure 7. Detecting and removing collisions between helices

In an intermediate step of PackHelix, two helices h_1 and h_2 have been placed too close: if the distance d between their respective axes h_1 and h_2 is smaller than a threshold set by the user, it is called a colli-sion. The collision is removed by displacing helix h_2 along a direction v until the minimum distance d is above that threshold. The moving direction v is computed as follows. The axis h_2 is projected on the sheet plane $\mathcal{S}$ defined by s_1 and s_2 (see figure 1 for details); v is a vector in the plane $\mathcal{S}$ that is perpendicular to the projection h_2', pointing in a direction away from h_1.

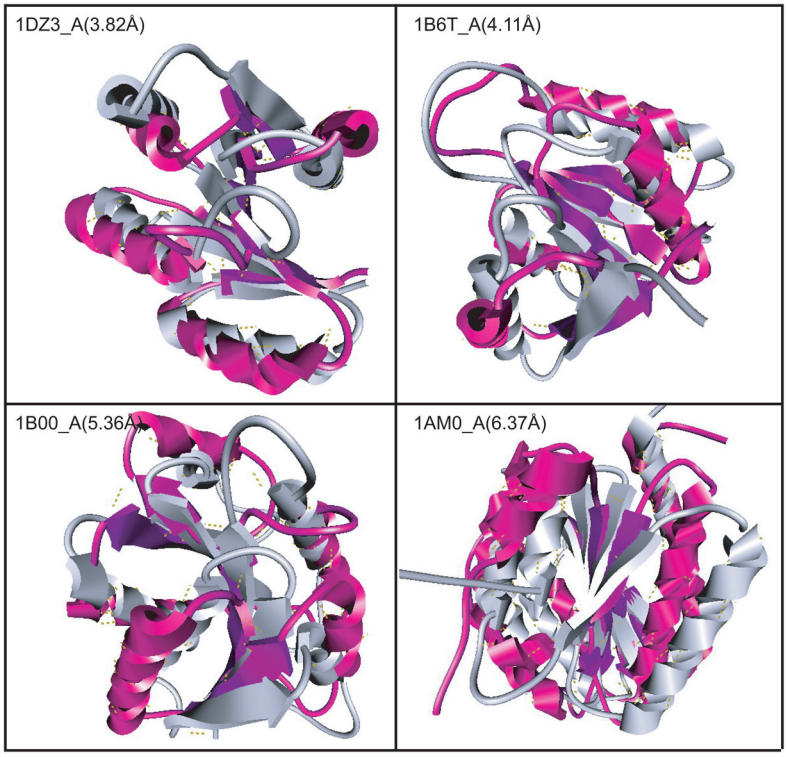

Figure 8. Results for test set 1

In test set 1, there is only a single helix between any two consecutive strands in the protein sequence. The 3D structures of our best models are rendered in gray color while the native structures are in purple and magenta colors. The RMSD in angstroms between the best model generated by PackHelix and the native structure is provided.

Figure 9. Test case 1H05

Superposition of our best model for the protein 3-dehydroquinate dehydratase (PDB ID 1H05 _A) (gray color) and the corresponding native structure (purple and magenta colors). The RMSD between the two structures is provided.

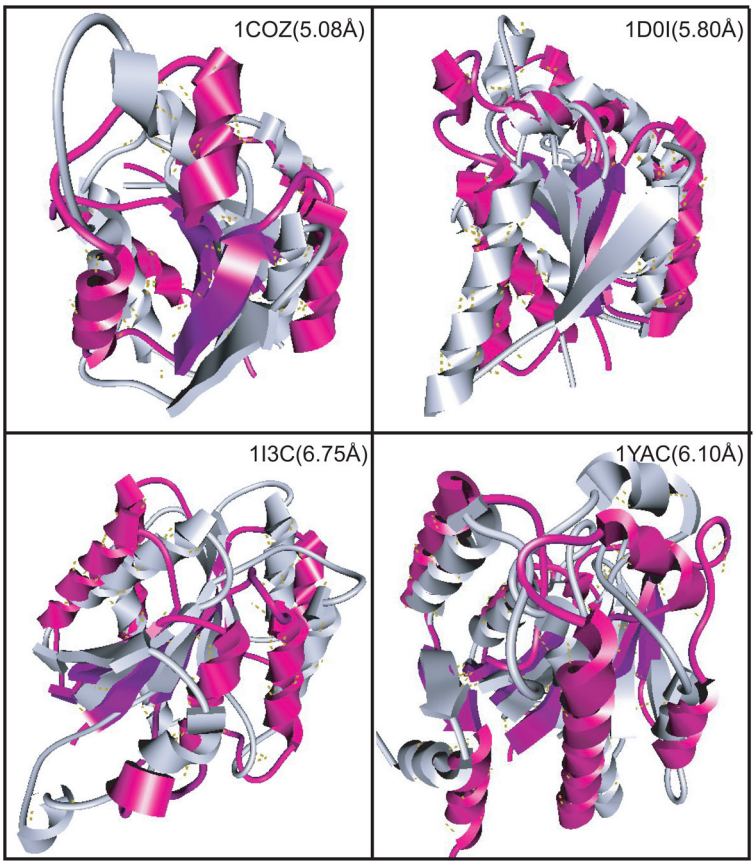

Figure 10. Results for test set 2

In test set 2, there is at least one occurrence of multiple helices between two consecutive strands in the protein sequence. The 3D structures of our best models are rendered in gray color while the native structures are in purple and magenta colors. The RMSD in angstroms between the best model generated by PackHelix and the native structure is provided.

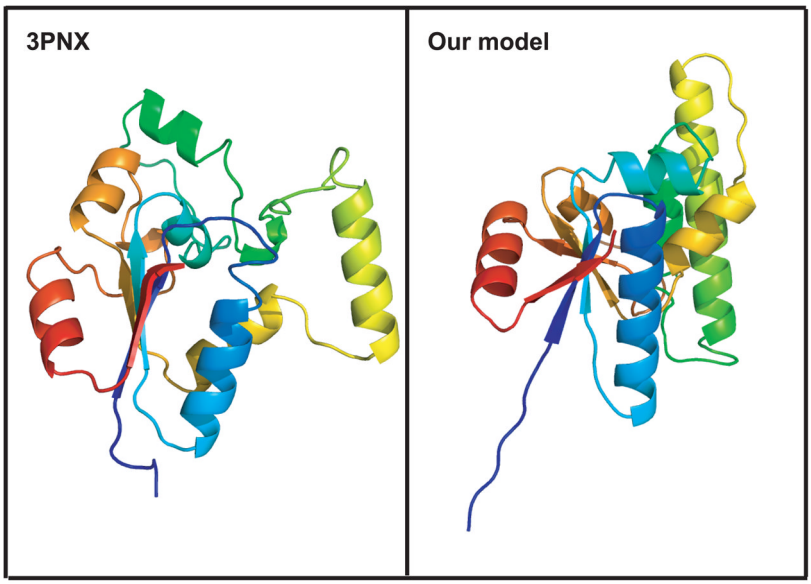

Figure 11. Modeling the target T0517 from CASP9

The model build with PackHe-lix that was submitted to CASP9 for the target T0517 (right panel) is compared with the corresponding native structure (PDB code 3PNX; left panel).

Table 1

Maximum stretched length vs number of residues for a loop

N	0P0U	0P1U	0P2U	1P0U	1P1U	1P2U	2P0U	2P1U	2P2U
4	16.8	13.8	10.3	13.9	12.5	9.3	11.2	7.9	6.8
5	19.9	17.9	11.8	14.7	14.2	10.6	14.0	11.9	9.1
6	21.1	20.7	13.7	20.7	19.5	12.7	17.3	15.2	10.7
7	24.2	25.2	15.3	23.7	22.2	14.6	23.7	23.0	12.3
8	28.3	27.2	16.9	28.1	25.7	15.8	27.7	25.5	13.7
9	31.5	32.3	22.7	31.0	30.4	19.7	31.0	28.9	17.8
10	35.6	35.5	24.6	35.4	34.1	23.0	35.0	34.1	21.2
11	38.7	39.4	28.3	38.2	37.6	27.1	38.3	37.6	25.6
12	42.8	42.6	30.5	42.6	41.1	29.9	42.2	41.0	28.3

N is the number of residues in a loop. U is the number of turnings along the path loop, while P defines the number of proline residues at the ends of the loop. The length is given in Å.

Table 2

Summary of results

PDB ID	Length	Number of Helices	Sampling (θ, ψ,D)[a]	Number of Structures [b]	Computing time(day:hour:minute) [c]	RMSD (Å) [d]
1DZ3 A	114	4	(3,2,1)	1621	00:09:20	3.82
1B6T A	119	4	(3,3,1)	3945	00:21:55	4.11
1B00 A	116	5	(3,3,1)	17015	03:22: 30	5.36
1AM0 A	172	5	(2,2,1)	875	00:04:50	6.37
1H05 A	135	8	(2,2,1)	626	00:05:15	4.68
1COZ A	113	6	(2,2,1)	741	00:06:10	5.08
1D0I A	144	10	(2,2,1)	559	00:04:35	5.80
1I3C A	130	6	(2,2,1)	468	00:03:55	6.75
1YAC	164	7	(2,2,1)	1667	00:13:50	6.10
T0517 (3PNX)	159	7	(2,2,1)	20	00:00:50	13.6

[a] Sampling numbers (n_t, n_p, n_D) chosen for the three measures θ, ψ, and D. The corresponding sampling values for θ are (165,145) if n_t = 2 and (164, 145, 120) if n_t = 3. Similarly, the sampling values for ψ are (5, −20) for n_p = 2 and (5, −20, 30) for n_p = 3. The sampling value for D is always 10.

[b] Final number of structures generated by PackHelix that have passed the collision test, with a collision threshold set at 15.

[c] Computing time on a desktop processor with a Intel Xeon 3.73GHz processor, on a single core.

[d] RMSD in angstroms between the best model generated by Packhelix and the native structure (C_α atoms only). For the CASP9 target T0517, RMSD between the model we deemed to be the best at the time of submission and the native structure.