Assembly of Protein Tertiary Structures From Secondary Structures Using Optimized Potentials

Trinh Xuan Hoang,1 Flavio Seno,2 Jayanth R. Banavar,3 Marek Cieplak,4 and Amos Maritan5

1The Abdus Salam International Center for Theoretical Physics (ICTP), Trieste, Italy
2INFN-Dipartimento di Fisica “G. Galilei,” Università di Padova, Padova, Italy
3Department of Physics, 104 Davey Laboratory, Pennsylvania State University, University Park, Pennsylvania
4Institute of Physics, Polish Academy of Sciences, Warsaw, Poland
5International School for Advanced Studies and INFM, Trieste, Italy

ABSTRACT We present a simulated annealing-based method for the prediction of the tertiary structures of proteins given knowledge of the secondary structure associated with each amino acid in the sequence. The backbone is represented in a detailed fashion whereas the sidechains and pairwise interactions are modeled in a simplified way, following the LINUS model of Srinivasan and Rose. A perceptron-based technique is used to optimize the interaction potentials for a training set of three proteins. For these proteins, the procedure is able to reproduce the tertiary structures to below 3 Å in root mean square deviation (rmsd) from the PDB targets. We present the results of tests on twelve other proteins. For half of these, the lowest energy decoy has a rmsd from the native state below 6 Å and, in 9 out of 12 cases, we obtain decoys whose rmsd from the native states are also well below 5 Å. Proteins 2003;52:155–165. © 2003 Wiley-Liss, Inc.

Key words: LINUS; protein structure prediction; perceptron learning; ab initio prediction; protein folding; simulated annealing

INTRODUCTION

The protein folding problem entails the determination of the secondary and tertiary structure of a protein based on the knowledge of its sequence.1,2 The tertiary structure determines the functionality of the protein and thus the prediction of the folded structure is a central challenge. The protein folding problem can be successfully tackled by following the LINUS model of Srinivasan and Rose.7–9

LINUS, which was introduced by Srinivasan and Rose,7–9 This model incorporates the most important structural features of proteins including the backbone representation, simple descriptors of the side-chain geometry, excluded volume interactions, interactions between sidechains, and hydrogen bonding. LINUS has been demonstrated to be valuable7–9 for protein structure prediction, particularly for determining the propensities of various segments to form particular secondary structures. This determination is done at a fixed moderate temperature that can be chosen optimally.10 This is because LINUS embodies an interplay between energy and entropy: helices form as a result of minimizing the energy whereas strands are favored by the entropy.

In this report, we build on LINUS in the context of ab initio tertiary structure prediction given a sequence of amino acids and their secondary structure. We demonstrate that LINUS has many features that are appealing for this task. First, based on an appropriate Monte Carlo dynamics, it is able to fold an extended chain to a globular form without violating any steric constraints. Second, the dynamics allow for an efficient exploration of the low-energy states in a way that singles out the native state provided one parametrizes the energy functions in a proper manner. We determine the parameters through a learning procedure.

As noted earlier, we simplify the task of tertiary structure prediction by assuming an a priori knowledge of the location of the secondary structures of the proteins under study. It should be noted that many of the current methods of proteins structure prediction rely, to various extents, on being preceded by prediction of the secondary structures. Indeed, by using a combination of techniques such as multiple sequence alignment and neural networks, one can predict the basic secondary structural features, e.g., α-helices and β-strands, with a confidence that exceeds...
Our simplified task could be thought of as a step in the ultimate goal of ab initio tertiary structure prediction. As we shall demonstrate, an important attendant advantage of such a simplified challenge is that it allows one to glean fundamental insights into the protein folding problem.

In this study, we adopt a simulated annealing procedure to study a LINUS-based model at low temperatures. In order to accommodate this shift in focus from intermediate to low temperatures, we work with a modified version of the LINUS model in which the original simplified interaction potentials are endowed with more amino acidic specificity. The basic idea of our learning procedure is to generate low-energy conformations, which preserve the pre-assigned secondary structure of proteins with known tertiary structure with a set of carefully chosen potential energy parameters. These low-energy conformations are then used as decoy conformations for carrying out a refinement of the potential energy parameters in order to ensure that the known native state structure is indeed lower in energy than the decoy conformations. This procedure, when iterated, leads to an estimation of the optimal potential energy parameters. Strikingly, our procedure can be used to assess whether the form chosen for the potential energy is adequate to the task of successfully discriminating between the true native state structure and the realistic competing decoy conformations.

Our studies lead to the conclusion that the potential energy parameters for a commonly used pairwise interaction can be optimized to simultaneously ensure that the native state structures of three distinct proteins can be successfully discriminated against decoy conformations. The parameters obtained by learning the three proteins provide an adequate description of these proteins: the LINUS native states are very close to the experimentally determined structures, the energy landscape is funnel shaped, steric clashes are avoided, and the native state is kinetically accessible. Furthermore, the parameters are also found to be adequate for several other proteins.

The Hamiltonian contains local and nonlocal terms. The former correspond to the fixed distance tethering by the peptide bonds and the torsional energies. The latter correspond to the hydrogen bonds, and to the pairwise contact interactions. A torsional energy is included to prevent formation of conformations with positive Ramachandran angles by associating a positive cost, $\epsilon_{\text{tor}}$, with them. The exception is glycine for which positive values of $\phi$ are rewarded with a negative energy of $-\epsilon_{\text{tor}}$.

Hydrogen bonds can be formed between the backbone N atoms and either the backbone O atoms (backbone-to-backbone H-bonds) or the sidechains of the amino acids from the set [Ser, Thr, Asn, Asp, Gln, Glu] (sidechain-to-backbone H-bonds). The distance between the donor and acceptor must be smaller than 5 Å and 4 Å for these two cases, respectively. It is also required that the dihedral angle between the O($k$)–N($l$)–C($d$($l$) plane and the N($l$)–C($d$)–C($l$ – 1) plane should be larger than 140°, where $k$ and $l$ are the indices of the amino acids involved along the sequence. The contact interaction arises between the sidechain atoms of the amino acids. A contact is declared to be formed if the distance, $r_{ij}$, between the two atoms, $i$ and $j$, is smaller than $R_i = R_j = 1.4$ Å, where $R_i$ and $R_j$ are the contact radii of the atoms. The energy of a contact decreases linearly from zero to its minimal value as the distance between the atoms decreases from $R_i$ to $R_i + R_j$, and it remains constant at smaller distances. This constant depends on the specificity of the sidechains. It should be noted that the contact radii of the atoms are larger than their hard sphere radii so that atoms can be fully in contact without steric clashes.

Our implementation of LINUS incorporates two crucial changes compared to the original formulation. The first is that we distinguish between the short-range backbone-to-backbone H-bonds that correspond to the H-bonds within helices and turns and the long-range backbone-to-backbone H-bonds (the ones that are further than 4 residues apart along the sequence). We have found that using a lower energy (more favorable) score for the long range H-bonds is essential for the assembly of the β-strands into a sheet.

The second change is an increase in the variety of the contact interactions. In the original LINUS, the amino acids were divided into three categories: hydrophobic, amphipathic, and polar. The contact interactions were allowed only between the hydrophobic and the amphipathic amino acids. We have verified that the interactions provided by considering just three kinds of amino acids are not enough to make the native state the global energy minimum. Thus, we consider 20 distinct types of amino acids. Because glycine, as defined above, does not participate in contact interactions, there are 190 different energy parameters corresponding to different contacts between the remaining 19 amino acids. A special case corresponds to the contact between the S atoms of two cysteines. A disulfide bond is said to be formed if two such S atoms are closer than 3.6 Å and the energy of the bond decreases linearly to its minimal value as the distance decreases to 2.65 Å. This energy remains constant when the distance
The sidechains can be rotated around the Ca-Cβ bond and the corresponding torsion angle χ may take any value. However, in order to enhance the acceptance rate the values related to the rotamer positions are chosen with a higher probability. Thus χ is chosen randomly in one of the three windows: $-60 \pm 20^\circ$, $60 \pm 20^\circ$, and $180 \pm 20^\circ$ with a probability of 0.9 and the values outside the windows are picked with a probability of 0.1. All other angles and bond lengths are kept fixed during the simulation except that the torsion angle ω about the peptide bond can vary within $180 \pm 10^\circ$, and the N-Cα-C bond angle can vary within $110 \pm 5^\circ$.

The simulation starts from an open conformation with the α-helices and β-strands built in, based on the experimental structure file from the Protein Data Bank (PDB). The moves proceed from the N-terminus and a complete progression to the C-terminus is called a cycle. A new conformation selected in this way is rejected if it leads to steric clashes (up to 50 attempts are made to decide on a move that does not generate steric clashes), otherwise it is accepted with a probability $P = \min(1, e^{-\Delta E/T})$, where $\Delta E$ is the energy difference compared to the previous conformation and $T$ is a fictitious temperature.

The low-energy conformations are obtained through an annealing scheme in which the temperature is decreased in steps according to the formula $T_{k+1} = 0.97T_k$, where $k$ is a step index. The starting temperature is chosen so that the acceptance rate, $r$, is larger than 0.1, and once $r$ falls below 0.001 we carry out a zero temperature quench. The typical length of a simulation is between 50,000 and 100,000 cycles.

**Perceptron Method for Learning the Interaction Potentials**

The principal requirement for the stability of the native state is that the native state has a lower energy than any other viable conformation. This requirement is also a necessary condition for the simulated annealing to lead to the native state from an open conformation. Thus

$$E(\Gamma_0) < E(\Gamma_D) \quad \forall D,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0,$$  

where $\Gamma_0$ denotes the native conformation and $\Gamma_D$ denotes the conformation of a decoy. We have found that, even with the secondary structures incorporated in the decoys, the conditions given in Eq. (1) cannot be satisfied by using unrefined interactions such as those used in the original version of LINUS.

The optimal stability perceptron algorithm is an iterative procedure that allows one to find a solution that satisfies a given set of linear inequalities optimally. We follow one of the several learning schemes discussed in Dima et al. and, in the context of determining environmental scores for the amino acids, Chang et al. For a given set of decoys and a known native conformation, our learning procedure deals with a set of inequalities of the type

$$\frac{E(\Gamma_D) - E(\Gamma_0)}{d(\Gamma_D, \Gamma_0)} > 0.$$
where $d(\Gamma', \Gamma)$ denotes the root mean square deviation (rmsd) of the decoy structure from the native state. The use of the rmsd in the denominator helps to shape the energy landscape into a funnel and ensures that a conformation with a large rmsd is less preferred than conformations that are more native-like. Figure 2 illustrates that our procedure, indeed, leads to the required features.

A brief description of the perceptron method is as follows. The inequalities (2) can be rewritten in the form

$$\hat{A}_k \cdot \hat{\epsilon} > 0, \quad k = 1, 2, \ldots, M,$$

where $\hat{\epsilon} = [\epsilon_1, \epsilon_2, \ldots, \epsilon_{194}]$ is a 194-component vector defined by the energy parameters to be optimized, $\hat{A}_k$ is a vector of the corresponding coefficients of the $k$-th inequality, $M$ is the total number of inequalities (the combined number of decoys for all proteins in the training set), and the dot denotes a scalar product. For a given training set of parameters $\hat{\epsilon}$, the stability of the $k$-th inequality is defined by $Q_k = \hat{A}_k \cdot \hat{\epsilon}$, while the perceptron stability is defined as $Q = Q_l$ with $l$ such that $Q_l < Q_k \forall k \neq l$. The stability $Q$ can be maximized by iteratively replacing $\hat{\epsilon}$ by a new vector given by $\hat{\epsilon}' = \hat{\epsilon} + \delta \cdot \hat{A}_q$, where $\delta$ is a small number. Each iteration step is followed by a normalization of $\hat{\epsilon}$ so that $\Delta = \sqrt{\sum_k \epsilon_k^2}$ remains constant while $Q$ is recomputed. A convergence is guaranteed to be reached in a finite number of steps. In our case, this typically takes tens of thousands of iterations and this number depends on how small the value of $\delta$ is. The starting values for $\hat{\epsilon}$ can be arbitrary. If $Q > 0$, the problem is learnable, which means that there exists a set of energy parameters such that all of the inequalities are satisfied. Indeed, if many solutions exist, the perceptron method selects the best among them.

Generation of LINUS Native Conformation

Since LINUS is an approximate model—it uses idealized bond angles and bond lengths for the backbone and a simplified representation of the sidechains—it cannot be expected to provide a perfect fit to the PDB structures. However, in order to make use of inequalities (2), in the learning process we need to find a LINUS-based conformation that plays the role of the native state. We determine it by arranging the backbone to have the PDB-derived values of $\phi$ and $\psi$ and then perform moves in order to reduce the rmsd to the PDB backbone as much as possible. In practice, we easily arrive at conformations with the backbone rmsd as small as 0.1 Å for the small size proteins considered here. Subsequently, the LINUS sidechains are built in and the sidechains are rotated in order to eliminate steric clashes. At this stage, the energy is minimized by the Monte Carlo quenching process without imposing any constraints on the segments that correspond to the secondary structures. The energies here are assigned treating the proteins as homopolymers, i.e., all of the energy parameters, but the torsional energy, are assumed to be uniform and negative. The final best native-like conformation usually has a rmsd of about 0.5–1.0 Å from the PDB structure.

RESULTS AND DISCUSSION

Learning the Interaction Potentials From Three Training Proteins

Three proteins have been chosen for learning the energy parameters: the B1 domain of protein G (the PDB code is 1GB1), ribosomal protein L7 (1CTF), and crambin (1CRN). They are all $\alpha/\beta$ proteins and their lengths do not exceed 70 residues. Crambin contains 3 disulfide bonds while the other two do not contain any cysteines. Their native conformations are shown in Figure 3 (left). For each protein, we have generated the LINUS-based native-like conformations using the fine-tuning procedure described previously. Their rmsd’s relative to the corresponding PDB structures are 0.56 Å, 0.68 Å, and 0.72 Å for 1GB1, 1CTF, and 1CRN, respectively.

The decoys for those proteins are generated with built-in constraints on the secondary structures. The identification of the secondary structures is obtained from the PDB files whenever it is available. For protein G, this kind of information is not present in the PDB file for 1GB1. In this case, we take the assignment of the secondary structure from another file, corresponding to 1PGA, which contains the X-ray determined coordinates for the same protein. The residues at the end of the $\alpha$-helices or $\beta$-strands can sometimes be confused with turns or coils. Therefore, we have effectively shortened some secondary structure fragments by one or two residues from their ends. An example of this situation is the second $\beta$-strand of 1CTF (residues 92–98 according to the PDB assignment), which is significantly distorted at positions 94, 97, and 98. For this case, we impose the strand geometry, in the Monte Carlo quenching procedure, which results in a conformation that is closer to the native state.
Carlo process, only on the residues from site 92 to 96. The α-helices usually show much smaller irregularities than the β-strands and the adjustments are much less severe. In general, adjustments are made in order to accommodate residues that lie too far from the Ramachandran regions associated with a given type of secondary structure. Such assignment changes have been made for 3 proteins in the learning set and 1 protein in the test set. Table I indicates all the modifications that we have made in the secondary structure assignment given in the PDB file. Note that the PDB information is needed only for α-helices and β-sheets and the records on turns and other type of secondary motifs such as the 3_10-helix are irrelevant for our calculations. It should be noted that, even with these irregularities removed, the native values of the φ and ψ angles for some residues in the secondary structures may still lie outside the expected regions in the Ramachandran plot within which they are dynamically constrained. This situation is illustrated in Figure 1 for the 1CTF protein. However, we have found that, for most of the proteins considered here, we can generate conformations within 3 Å in rmsd from the PDB structures with the constraints imposed on the secondary structures.

The learning process is implemented in an iterative manner by starting from some generic potentials (as derived from Srinivasan and Rose[7,8]) and generating the corresponding decoys. We identify decoys that complete with the native state in that they have energies comparable to or better than the native state energy. We adjust the energy parameters to destabilize the decoys. At the next stage, we use the just learned potentials and generate new corresponding decoys, and so on. The set of decoys keeps expanding through accumulation and all of them are used to generate the inequalities. Furthermore, the inequalities derived for different proteins are combined together. Decoys that are found to be within 2 Å from the PDB structure are discarded because these conformations are too close to the native state to provide unbiased inequalities that could properly shape the folding funnel (the number of such discarded decoys is typically one or two for a given protein). In total, we have generated approximately 1,200, 1,200, and 500 decoys for the 1GB1, 1CTF, and crambin proteins, respectively. In the final rounds of learning, approximately 50 new decoys are generated for each protein. We stop when the predicted structures in the training set have a rmsd that is smaller than 3 Å.

Figure 4 shows the plot of energy vs. rmsd for all the decoys for 1GB1 using the energy parameters that have been learned at the final round of learning. It can be seen that the decoys cluster in a way that suggests an emergence of a funnel-like energy landscape. The energy decreases with the rmsd at the lower boundary of the cluster, as indicated by the dashed line in Figure 4, and the native state is the global energy minimum. The perceptron stability Q is the slope of the dashed line. Note that the bigger the Q, the more funnel-like the energy landscape. On the other hand, the learning process ought to result in a systematic decrease of Q because more and more inequalities are taken into account unless convergence is achieved. This can be seen in Figure 5, which shows Q/Δ as a function of the number of decoys where Δ = √Σp_i^2, the square root value of the energy parameters, provides a normalization. Q/Δ is seen to decrease with the number of decoys used for learning. A sharp decrease at around 2,400 decoys corresponds to arriving at a situation at which the set of decoys is sufficiently large to provide a significant reduction in the size of the relevant parameter space. After this decrease, there is a plateau in Q, which suggests a convergence of the energy parameters.

On optimizing the energy parameters, as described above, our annealing procedure is able to predict the tertiary structures for the three proteins that were used in learning with a very good accuracy. Among about 50 decoys generated for each of the three proteins after the learning was completed, the lowest energy state is found to be below 3 Å in rmsd from the PDB structures (see Table I and Fig. 3). Furthermore, as shown in Figure 4 for 1GB1 and 1CTF, the decoys appear to be in a well-developed funnel-shaped energy landscape. Figure 3 compares the
PDB native conformations to the predicted conformations. Although the rmsds seem to be still somewhat large, e.g., 2.94 Å in the case of 1CTF, the predicted conformations exhibit striking similarities to the PDB structures. The best success in prediction is for 1GB1, for which the rmsd of the lowest energy decoy is 1.90 Å away from the PDB structure.

After optimizing the potentials, we rescaled our parameters to set the energy of a short range backbone-backbone H-bond equal to \(-1.0\). In these units, the energy of a long-range backbone-backbone H-bond is found to be \(-1.4952\) and the sidechain-backbone H-bond has a value of \(-0.7779\). In contrast, the original LINUS parameters for these two situations were equal to and twice the

---

**TABLE I. Name, PDB Code, Length, and Class of the Proteins Studied and the rmsds of the Decoys Obtained by Our Prediction Using the Optimized Potentials**

<table>
<thead>
<tr>
<th>Name</th>
<th>PDB code</th>
<th>Length</th>
<th>Class</th>
<th>Lowest rmsd (Å)</th>
<th>Lowest energy decoy (Å)</th>
<th>Most tube-like decoy (Å)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protein G(^a)</td>
<td>1GB1</td>
<td>56</td>
<td>α/β</td>
<td>1.90</td>
<td>1.90</td>
<td>4.99</td>
</tr>
<tr>
<td>Ribosomal protein L7(^b)</td>
<td>1CTF</td>
<td>68</td>
<td>α/β</td>
<td>2.94</td>
<td>2.94</td>
<td>2.94</td>
</tr>
<tr>
<td>Crambin(^c)</td>
<td>1CRN</td>
<td>46</td>
<td>α/β</td>
<td>2.04</td>
<td>2.15</td>
<td>3.68</td>
</tr>
<tr>
<td>Zinc-finger(^d)</td>
<td>1ZAA</td>
<td>28</td>
<td>α/β</td>
<td>2.11</td>
<td>2.98</td>
<td>3.28</td>
</tr>
<tr>
<td>Protein L(^e)</td>
<td>1HZ6</td>
<td>62</td>
<td>α/β</td>
<td>4.03</td>
<td>4.03</td>
<td>16.58</td>
</tr>
<tr>
<td>Cro repressor</td>
<td>1ORC</td>
<td>64</td>
<td>α/β</td>
<td>4.79</td>
<td>4.79</td>
<td>6.00</td>
</tr>
<tr>
<td>Merp</td>
<td>1AFI</td>
<td>72</td>
<td>α/β</td>
<td>4.96</td>
<td>6.83</td>
<td>10.37</td>
</tr>
<tr>
<td>E. coli Reca</td>
<td>1AA3</td>
<td>63</td>
<td>α/β</td>
<td>8.17</td>
<td>8.17</td>
<td>10.71</td>
</tr>
<tr>
<td>Hyperthermophile Sac7D</td>
<td>1SAP</td>
<td>66</td>
<td>α/β</td>
<td>8.39</td>
<td>11.91</td>
<td>11.58</td>
</tr>
<tr>
<td>Chea</td>
<td>1FWP</td>
<td>69</td>
<td>α/β</td>
<td>9.29</td>
<td>12.09</td>
<td>12.29</td>
</tr>
<tr>
<td>Cole1 ROP protein</td>
<td>1RPO</td>
<td>61</td>
<td>α</td>
<td>3.37</td>
<td>4.63</td>
<td>4.43</td>
</tr>
<tr>
<td>Human P8-MTCP1</td>
<td>1HP8</td>
<td>68</td>
<td>α</td>
<td>4.14</td>
<td>5.74</td>
<td>9.87</td>
</tr>
<tr>
<td>Staphylococcal protein A(^f)</td>
<td>1BDD</td>
<td>52</td>
<td>α</td>
<td>3.67</td>
<td>5.85</td>
<td>8.66</td>
</tr>
<tr>
<td>Pheromone ER-1</td>
<td>1ERC</td>
<td>40</td>
<td>α</td>
<td>4.38</td>
<td>7.89</td>
<td>8.29</td>
</tr>
<tr>
<td>434 repressor</td>
<td>1R69</td>
<td>63</td>
<td>α</td>
<td>4.21</td>
<td>11.31</td>
<td>6.30</td>
</tr>
</tbody>
</table>


\(^c\)Secondary structure assignment: helices (7–17, 23–30), strands (1–4, 32–35).

\(^d\)Secondary structure assignment: helices (7–17, 23–30), strands (1–4, 32–35).

\(^e\)Secondary structure assignment: helices (33–60).


\(^g\)Columns showing the lowest rmsd, the rmsd of the lowest energy decoy, and the rmsd of the decoy were selected by using a criterion based on the tube picture of protein structures (see text). The first three proteins have been used for learning, and the prediction is made for the other 12 proteins in the list. The secondary structure assignment is taken as that given in the PDB file for all proteins, except the ones marked with superscripts a, b, c, and e. The assignments used for these proteins are indicated.

---

**Fig. 4.** Energy vs. rmsd for decoys generated using the optimized potentials for 1GB1. The open circle indicates the LINUS conformation used as the native state in the learning. **Inset:** A similar plot for 1CTF.

**Fig. 5.** The perceptron stability, \(Q\), normalized by the square root of energy parameters, \(\sqrt{\sum \frac{E_i}{\delta^2}}\), plotted as a function of the number of decoys, \(M\), used in the learning procedure.
short-range backbone-backbone H-bond, respectively. The
torsion energy remains positive and the hydrophobic inter-
actions are better differentiated. However, we find that, on
average, the contact energies between groups of sidechains
diff of different hydrophobicity are in rough accord with the
scale introduced by Srinivasan and Rose.7,8

Prediction of Tertiary Structures Using the
Optimized Potentials

Once the optimized potentials from three training pro-
tein structures were determined, we carried out prediction of the
tertiary structures for a dozen additional proteins assum-
ing knowledge of the secondary structure. These proteins
are listed in Table I. Seven of them are of the α/β type and
five of the α type. Their lengths vary between 28 and 72.
Fifty decoys are generated for each protein and the one
that was lowest in energy was taken as a representation of
the native state. The results of our simulations are com-
pared in Table I, in which we present the rmsd of the lowest
energy decoy with respect to the PDB target and also the
lowest rmsd among all the decoys obtained through the
simulated annealing. It can be seen that we are able to
predict the tertiary structure with an accuracy of below 6 Å
for half of the new proteins, and in the four best cases the
rmsd of the predicted conformation is below 5 Å. We have
checked that if we use the standard LINUS energy param-
eters without the adjustments obtained through learning
then the predicted structures are further than 8 Å away from
the targets.

Figure 6 compares our predictions to the PDB targets for
three α/β-proteins: the zinc-finger (1ZAA), protein L (1HZ6),
and cro repressor (1ORC). Although the rmsds obtained
are not as good as for the training proteins, the close
similarities between the predicted and targets structures
can be easily recognized. The overall topologies are correct
in all cases. The best prediction in terms of the rmsd
belongs to the zinc-finger, which is also the smallest
protein in the set. However, there are some noticeable
flaws in the predicted structure. The first one is due to the
irregularity of the single helix in the PDB structure. The
N-terminal part of this helix is twisted in a tighter way than the regular helix. Our procedure fails to reproduce
the correct twist because the Ramachandran regions used
for the helices correspond only to perfect helices. The
second flaw is that the β-hairpin is not as close to the helix
as it should be. This can be due to the presence of a zinc
atom, which is covalently bonded to the sidechains of two histidines from the helix and two cysteines from the
β-hairpin in the real structure. These bonds stabilize the
structure significantly but they are not taken into account
in our model.

The second best predicted protein in the set is protein L.
Its native topology is close to that of protein G, which has
been used for learning. However, protein L is 6 residues
longer and shares little sequence similarity with protein G
and yet our procedure is able to predict the tertiary
structure with a correct topology and a rmsd of nearly 4 Å.
The cro repressor is also an interesting case. The forma-
tion of the β-sheet and the relative positions of the helices
are very well predicted. The relatively large rmsd of 4.79 Å
is primarily due to poor prediction of the loops at the two
terminals.

The prediction is bad for four α/β-proteins in the set: the
mercuric transport protein (1AFI), the C-terminal domain
of the Escherichia coli reca (1AA3), hyperthermophile
protein (1SAP), and the chyebinding domain of chea
(1FWP). For these proteins, the lowest rmsd and the rmsd
of the lowest energy decoy are all larger than 8 Å from the
PDB targets. This indicates that the learned potentials are
not adequate for those proteins. Another possibility is that
their native conformations depend on the binding cofac-
tors, which are not taken into account in our simulations.
Note that both 1AA3 and 1SAP bind to DNA, whereas
1FWP binds to another substrate.

The predictions for the α-proteins are also reasonably
good even though the potentials were learned using α/β-
proteins. For the ROP protein (1RPO), a two-helix bundle
with a simple topology, we obtained the lowest energy
decoy with a rmsd of 4.63 Å. For the next three-helix
bundles: human P8MTCP1 (1HP8), staphylococcus pro-
tein A (1BDD) and pheromone ER-1 (1ERC), the predicted
rmsds are 5.74 Å, 5.85 Å, and 7.89 Å, respectively. For the
five-helix bundle 434 repressor (1R69), the prediction is a failure since the lowest energy decoy is 11.31 Å away from the native state. It should be noted that for all helical proteins the lowest rmsd obtained is quite low (in the range from 3.3 Å to 4.4 Å) in comparison to some α/β proteins. This is due to the fact that the α-helix bundles are easily accessible dynamically.

It should be noted that the overall rmsd is not the only measure for the assessment of the predictions. As we have seen in Figure 6 for Cro repressor, the large rmsd, 4.79 Å, of the predicted structure primarily arises from the misfolding of the two tails of this protein, while the rest is correctly formed. Another example is the case of the hyperthermophile protein (1SAP) shown in Figure 7. For this protein, in the predicted structure, the helix is misfolded and it causes a very large overall rmsd of 11.9 Å. If one considers only a fragment of this protein, which excludes the helix, the rmsd of the predicted structure with respect to the native structure is reduced to 4.89 Å. For this fragment, the N-terminal α-hairpin and the central β-sheet with three strands are correctly formed.

In order to assess the quality of our predictions, in Figure 8 we plot the rmsd vs. the length of the fragment that was predicted the best when compared to the PDB target. The solid lines are for the three proteins that were used in the learning procedure, the dashed lines are for α/β proteins from the testing set, and the dotted lines are for the α proteins. The thicker dashed line corresponds to 1SAP.

Fig. 7. The structure prediction for 1SAP. The darker regions show the first 50 residues for which the rmsd is 4.89 Å. The overall rmsd is 11.91 Å because the helix is misfolded.

Fig. 8. The rmsd vs. the length of the fragment that was predicted the best when compared to the PDB target. The solid lines are for the three proteins that were used in the learning procedure, the dashed lines are for α/β proteins from the testing set, and the dotted lines are for the α proteins. The thicker dashed line corresponds to 1SAP.

The proteins studied here. Except for 1CRN, the low rmsd structures do not correspond to peaks in the histograms. This suggests that the annealing used here may be too fast to avoid large rmsd conformations. On comparing the histograms of different proteins, one finds that for some proteins, such as 1GB1, 1CRN, 1ZAA, and 1RPO, the native state or more precisely the low rmsd conformations are more easily kinetically accessible unlike others such as 1SAP and 1FWP. Poor accessibility to the native state could arise for different reasons including a poor annealing scheme, inefficiency of the dynamics, bad energy potential parametrization, or from factors intrinsic to the geometry of the native conformation itself. Interestingly, we are able to obtain low rmsd conformations consistently for α-proteins.

It should be noted that, for each of the three proteins in the learning set and a large number of the proteins in the test set, the lowest rmsd we obtain is of the same order as that found using Baker’s ROSETTA method. Note that ROSETTA uses predicted secondary structures whereas we use experimentally determined ones. Even though the total number of decoys generated for each protein in our study is much smaller than in ROSETTA, we are able to obtain lower rmsd decoys in several cases. It is interesting to note that for the two proteins, 1SAP and 1FWP, for which our scheme performs most poorly, ROSETTA was also unable to generate decoys with rmsds lower than 6 Å.

In none of the tested proteins, the native-like LINUS conformation has the lowest energy compared to the energies of the decoys. Thus, any poor predictions are not due to a lack of dynamical accessibility of the native state but due to its poor thermodynamic stability. One might expect that one ought to be able to improve the potentials and, thence, the thermodynamic stability by enlarging the training set. However, we have found that special care
should be taken when choosing proteins for learning and also a good preparation of the LINUS-based native conformation is needed. This is due to the nature of the perceptron learning for which one bad inequality may lead to a destruction of the entire set of potential energy parameters.

It has been recently pointed out that the protein backbone in the native state behaves like a tube with a strikingly uniform radius.\textsuperscript{27,28} This reflects special properties of the structural and chemical components of the protein sequence and the nature of its compact state. Thus, it is interesting to check whether the decoys generated by our methods behave like a tube and whether one can use this criterion to select the structure that is most protein-like in this respect. Given the Ca backbone of a decoy, the local radius of the tube at position $i$ is defined by

$$R_i = \min_{j,k}(R_{ijk}),$$

where $R_{ijk}$ is the radius of the circle drawn through 3 points given by the positions of the Ca atoms of indices $i$, $j$, and $k$ in the sequence. It has been found that in real protein structures $R_i$ has a very small fluctuation around 2.7 Å. In order to take this into account we define a tube parameter, which is given by

$$R_0 = \frac{1}{L} \sum_{i=1}^{L} (R_i - 2.7)^2,$$

where $L$ is the number of residues in the protein. The $R_0$ values for protein structures are typically smaller than 0.02. It is found that many of our decoys have $R_0$ significantly larger than this PDB value, but the best (lowest...
most important aspects of protein folding: the belief that proteins fold to their native conformations by searching for the global minimum in the energy landscape. With a priori knowledge of the secondary structures, proper folding is obtained for all three proteins used in the training set for which the potentials have been optimized. Despite some limitations, the transferability of the potentials to other proteins is shown to work reasonably well.

The energy function that we employed is rather simple with, basically, only two kinds of non-local interactions: hydrogen bonding and side-chain contact interactions. The success of the methods tested here indicates that these interactions are of primary importance in folding. Our work demonstrates that it is possible to fold a protein with pre-formed secondary structures without violating steric constraints in a well-defined microscopic model. This supports a scenario for the folding mechanism in which the secondary structures are formed early during the folding process.

ACKNOWLEDGMENTS

We are indebted to Raj Srinivasan and George Rose for their very helpful advice on LINUS. We also thank Giuseppe Zanotti for helpful discussions.

REFERENCES