PROTEINS: Structure, Function, and Genetics 49:154–166 (2002)

Combining the GOR V Algorithm With Evolutionary Information for Protein Secondary Structure Prediction From Amino Acid Sequence

A. Kloczkowski,[1] K.-L. Ting,[1] R.L. Jernigan,[1†] and J. Garnier[2]
[1]*Laboratory of Experimental and Computational Biology, NCI, NIH, Bethesda, Maryland*
[2]*Mathematical and Statistical Computing Laboratory, CIT, NIH, Bethesda, Maryland*

ABSTRACT **We have modified and improved the GOR algorithm for the protein secondary structure prediction by using the evolutionary information provided by multiple sequence alignments, adding triplet statistics, and optimizing various parameters. We have expanded the database used to include the 513 non-redundant domains collected recently by Cuff and Barton (Proteins 1999;34:508–519; Proteins 2000;40:502–511). We have introduced a variable size window that allowed us to include sequences as short as 20–30 residues. A significant improvement over the previous versions of GOR algorithm was obtained by combining the PSI-BLAST multiple sequence alignments with the GOR method. The new algorithm will form the basis for the future GOR V release on an online prediction server. The average accuracy of the prediction of secondary structure with multiple sequence alignment and full jack-knife procedure was 73.5%. The accuracy of the prediction increases to 74.2% by limiting the prediction to 375 (of 513) sequences having at least 50 PSI-BLAST alignments. The average accuracy of the prediction of the new improved program without using multiple sequence alignments was 67.5%. This is approximately a 3% improvement over the preceding GOR IV algorithm (Garnier J, Gibrat JF, Robson B. Methods Enzymol 1996;266:540–553; Kloczkowski A, Ting K-L, Jernigan RL, Garnier J. Polymer 2002;43:441–449). We have discussed alternatives to the segment overlap (Sov) coefficient proposed by Zemla et al. (Proteins 1999;34:220–223). Proteins 2002;49:154–166.**

Key words: GOR algorithm; protein secondary structure; secondary structure prediction; PSI-BLAST; multiple sequence alignment; information theory

INTRODUCTION

The prediction of protein structure, and ultimately protein function from amino acid sequence is arguably one of the most important problems in molecular biology. Recently, his problem has become dramatically more important, with completion of many large-scale genome sequencing projects yielding an enormous amount of amino acid sequence data, much of it corresponding to proteins of unknown function. The gap between the number of known protein sequences and the number of known structures in the Protein Data Base (PDB) continuously grows at an incredible rate. Some methods, such as homology modeling or threading are useful, but sometimes unfeasible, making major advances in protein structure prediction from sequence of the utmost importance. Although the prediction of tertiary structure is one of the ultimate goals of protein science, the prediction of secondary structure from sequence is still a more feasible intermediate step in this direction. Furthermore, some knowledge of the secondary structure can serve as an input for prediction.

Instead of predicting the full three-dimensional structure, it is much easier to predict simplified aspects of structure, namely the key structural elements of the protein and the location of these elements not in the three-dimensional space but along the protein amino acid sequence. This reduces the complex three-dimensional problem to a much simpler one-dimensional problem. The fundamental elements of the secondary structure of proteins are α-helices, β-sheets, coils, and turns. All these elements can be easily observed in the crystal three-dimensional structure of proteins in the PDB. Because such visual observation is rather subjective, there is need for a more rigorous definition of various elements of the protein secondary structure from the atomic coordinates in the PDB. In 1983, Kabsch and Sander[1] developed the classification of elements of secondary structure based mainly on hydrogen bonds between the backbone carbonyl and NH groups. Their dictionary of secondary structure assignment Database of Secondary Structure in Proteins (DSSP) is widely used in protein science (although there are other alternative assignment methods, such as STRIDE) and the DSSP server was established at European Molecular Biology Laboratory (EMBL) in Heidelberg with all proteins in the PDB bank given DSSP assignments.[2] According to the DSSP classification, there are eight elements of secondary structure assignment denoted by letters: H (α-helix), E (extended β-strand), G (3_{10} helix), I (π-helix), B (bridge, a single residue β-strand), T (β-turn),

†Correspondence to: R.L. Jernigan, Laurence H. Baker Center for Bioinformatics and Biological Statistics, Iowa State University, Ames, IA 50011-3020. E-mail: jernigan@iastate.edu

Received 20 December 2001; Accepted 30 April 2002

S (bend), and C (coil). Coil is defined as a structural element that does not belong to any of the other seven classes. Eight types of the secondary structure are too many for existing methods of the secondary prediction, instead usually only three states are predicted: helix (H), extended (β-sheet) (E), and coil (C). Instead of coil (C), some authors use the nomenclature nonregular, aperiodic structure or a loop with the abbreviation (L).[3] The eight-letter DSSP alphabet requires translation into the three-letter code. For instance, for the CASP[4] (Critical Assessment of Structure Prediction) experiments, helices (H, G, and I) in the DSSP code are assigned the letter H in the three-letter secondary structure code, whereas strands (E) and bridges (B) in the DSSP code are translated into sheets (E) in the three-letter code. Other elements of the DSSP structure (T, S, C) are treated as coil (C). There are, however, other ways to make these assignments. For some authors, I (π-helix) is translated into coil (which is not so important because of the rarity of I structures). Frishman and Argos[5] assumed that the DSSP H and E are translated to H and E in the three-state code, and all other letters of the DSSP code are translated to coil (C). Additionally, they assumed that helices shorter than five residues (HHHH or less) and sheets shorter than three residues (EE) are coils. In this article, we adopt the Frishman and Argos translation of the DSSP alphabet into the three-code to calculate the accuracy of prediction of the secondary structure by the GOR algorithm. Some other authors have treated helical DSSP G elements as helices H in the three-letter code only if they were neighbors to H sequences, but isolated G elements were treated as coil (i.e., GGGHH-HHH is translated to HHHHHHHHH whereas CCCGGGCCC is translated to CCCCCCCCC).[6,7] The bridge (B) is the DSSP structure that is most difficult for the secondary structure prediction, because it is only one residue long. This is the reason that frequently the corrections for bridges B are done, namely the sequence BC is translated to EE and BCB to CCC.[3] The GOR program uses the correction algorithm which removes all such short secondary structure sequences as the most likely assignment errors. To add to the confusion, the DSSP assignment algorithms are not all identical. The DSSP algorithm used in the PDB differs from the original DSSP algorithm developed at EMBL, and their assignments differ slightly. The original EMBL algorithm takes into account inter-chain hydrogen bonds, which the PDB algorithm does not. Additionally, hydrogen bond placement can be different because of ambiguous interpretation of imperfect geometries inherent in experimental structures (Philip Bourne, personal communication).

It is worth noting that the theoretical problem of one-dimensional secondary structure prediction with three states (H, E, C) can be reduced even further to only two states by defining the structural states of residues with respect to their solvent accessibility, with compact globular proteins having residues either buried inside and inaccessible to water, or exposed residues on the surface and easily accessible to water.[3] This binary classification of structural elements of proteins corresponds to the simple hydrophobic-polar (HP) model of proteins. Such simple models are sometimes useful to localize protein functional sites and predict the functional properties of a given amino acid sequence, such as trans-membrane proteins.

The CASP experiments initiated by John Moult,[4] which since 1994 have taken place every second year, gave the protein structure prediction a new impetus. Scientists from different laboratories try to predict structures of newly discovered but not yet published crystallographic data of new PDB entries. The experiment helps to assess and compare various prediction algorithms and techniques, and the current state of the art of prediction methodology.

To measure quality of protein secondary structure prediction, it is convenient to introduce an accuracy matrix $[A_{ij}]$ of the size 3×3 (i and j stand for the three states H, E, C). The ij-th element A_{ij} of the accuracy matrix is then the number of residues predicted to be in state j, which according to the DSSP data are actually in state i. Then the sum over the columns of matrix A gives the number of residues n_j that are predicted to be in state j[3]:

$$n_j = \sum_{i=1}^{3} A_{ij} \tag{1}$$

However, the sum over the rows of A gives the number of residues N_i which according to the experimental data are in state i:

$$N_i = \sum_{j=1}^{3} A_{ij} \tag{2}$$

It is obvious that the diagonal elements of A count the correct predictions for each of three structural states, and the off-diagonal elements contain the information about wrong predictions.

The main parameter measuring the accuracy of the protein secondary structure prediction is the parameter Q_3 defined as:

$$Q_3 = \frac{\sum_{i=1}^{3} A_{ii}}{N} 100 \tag{3}$$

which gives the percentage of all correctly predicted residues within the three-state (H, E, C) classes. Here N is the total number of residues in the sequence.

$$N = \sum_{i=1}^{3} N_i = \sum_{j=1}^{3} n_j \tag{4}$$

There are also parameters measuring individually the correctness of prediction for each of the structural classes such as:

$$q_i = \frac{A_{ii}}{N_i} 100 \quad \text{for } i = H, E, C \tag{5}$$

Usually, the easiest to predict are coils (C), then helices (H), and the most difficult for prediction are sheets (E).

For n states, the equally probable purely random assignment gives probability 100/n% of correct prediction, that is, for three states (H, E, C), the prediction less than 33.3% is worse than random assignment, if all three states were equally populated (which is not true for real proteins).

The parameter measuring the quality of prediction is the correlation coefficient C_i proposed by Matthews[3,8]:

$$C_i = \frac{A_{ii}\left(\sum_{k\neq i}^{3}\sum_{j\neq i}^{3} A_{jk}\right) - \left(\sum_{j\neq i}^{3} A_{ij}\right)\left(\sum_{j\neq i}^{3} A_{ji}\right)}{\sqrt{\left(A_{ii} + \sum_{j\neq i}^{3} A_{ij}\right)\left(A_{ii} + \sum_{j\neq i}^{3} A_{ji}\right)\left(\sum_{k\neq i}^{3}\sum_{j\neq i}^{3} A_{jk} + \sum_{j\neq i}^{3} A_{ij}\right)\left(\sum_{k\neq i}^{3}\sum_{j\neq i}^{3} A_{jk} + \sum_{j\neq i}^{3} A_{ji}\right)}} \tag{6}$$

which allows us to compare the result of the prediction with the completely random assignment. For the perfect prediction, the Matthews coefficient $C_i = 1$ whereas for completely random $C_i = 0$ (negative values of C_i are also possible, for predictions worse than random). Other quantities used for the assessment of the prediction accuracy are the average $\langle L_i \rangle$ length of each type of structural elements: helices, sheets and coils, the number of various secondary structure elements in the protein, and the segment overlap coefficients.[6,9,10] The number of the structural elements and predicted average lengths and their overlap should be close to experimental data for good predictions. On average, proteins contain about 30% helical structure (H), about 20% β-strands (E), and about 50% coil (C) structure. This means that even the most trivial prediction algorithm which assigns all residues to the coil (C) state would give approximately 50% correct prediction. The coil is also the easiest to predict, whereas strands (E) are the most difficult for prediction. The difficulty of the prediction of β-sheets is attributable to their relative rarity and to the irregular, nonlocal nature of contacts, in contrast to α-helices where contacts are well localized (the i-th and i + 4-th residue have a nonbonded contact).

Recently, it has been emphasized that the correctness of the prediction for individual residues has to be completed by the secondary structure overlaps.[6,9,10] A new properly normalized measure of this overlap, the so-called segment overlap (Sov_{obs}) was defined recently by Zemla et al.[9]

The first serious attempts of the secondary structure prediction which started in the 1970s with the seminal works of Chou and Fasman,[11] Lim,[12,13] and Garnier et al.[14] (GOR I method) were based on single sequences and gave the cross-validated accuracy of the prediction below 60%. All early works on the prediction of the secondary structure relayed on the single residue statistics in various structural elements. The predictions were done by using a sliding window of a certain size (for example of a width of four residues, a characteristic length for helical contacts, in the Chou and Fasman method,[11] or of width 17 residues in the GOR I method[14]) but only single residue statistics for each residue within such a window were calculated for the prediction. This gave a serious deficiency for these predictions.

A significant improvement in protein secondary prediction was done by using the pair-wise statistics for blocks of residues in secondary structure segments within the window (GOR III–IV). The practical implementation of this method is based also on a window of a certain width, which is moved along the protein chain. Then the statistics of the residues within the window are used to predict the conformational state of the residue at the center of the window. While the window moves along the chain, the secondary structure states of all residues from the N-terminal to the C-terminal along the chain are predicted.

This window-based method has been used by many different secondary structure prediction methods, based on various techniques, such as information theory (GOR III[15,16] GOR IV[17]), neural networks,[5,18–24] nearest-neighbor algorithms,[6,7,25–29] and several other approaches.[6,30–35] The accuracy of the prediction of these methods based on a single-sequence analysis has been improved significantly, breaking the 60% level but below the 70% limit for the most successful methods.

In the last few years, major progress has been made in the accuracy of the prediction of secondary structure from sequence (see review in Ref. 36). The improvement has been obtained by using, instead of a single sequence, multiple sequence alignments containing the evolutionary information about protein structure. The multiple sequence alignment information was used first in 1987 by Zvelebil et al.[37] and later (1993) supplemented by Levin et al.[38] and independently by Rost and Sander[18] for the prediction of secondary structure. It gave a significant boost to the accuracy of secondary structure prediction. The most successful methods like PHD[39] and its most recent versions, or PSIPRED[24] claim to achieve a prediction accuracy above 76%. Recently, Petersen et al.[23] announced the accuracy of predictions above 77% by using a neural network based prediction program. (The title of their article even contains the information about 80% accuracy of the prediction, but the actual cross-validated accuracy is about 77%.[36])

The main reason that information from the multiple sequence alignments improves the prediction accuracy is attributable to the fact that during evolution protein structure is more conserved than sequence, which consequently leads to the conservation of the long-range information. One may suppose that part of this long-range information is revealed by multiple alignments. Many proteins have a similar structure whereas having sequence identity as low as 20%. Protein function is more vital for evolutionary survival, than is sequence conservation so random mutations to the sequence that destroy its function usually cause the mutated sequence to be eliminated during evolution and hence do not exist.

The new efficient multiple sequence alignment programs such as PSI-BLAST[40] allow for easy use of the alignment information for secondary structure prediction. Multiple sequence alignment enables identification of the evolutionarily conserved residues and leads to improvement in the prediction of secondary structure.[36,41–43] Our recent results[44] show that many sequences having 25–30% identity compared with the query sequence, have their secondary structure predicted better than the query sequence. The results presented in this article further support this observation. Our computations indicate that small improvements in the accuracy of the prediction are even obtained by removing sequences from the multiple sequence alignment that are too similar to the query sequence, whereas removing the sequences with extremely low identity does not improve the secondary structure prediction.

The methodology of the secondary structure prediction is usually based on having a database of sequences with known secondary structure. The protein secondary structure prediction program finds relations between a sequence and structure by using the sequences in the database and using them for the prediction of the secondary structure of new sequences, different than those in the database. Consequently, the success of predictions depends to a large extent on the proper choice of sequences for the database. The database should cover all types of proteins in the most representative way, and no proteins in the database should be too similar. It is usually required that the similarity between the sequences in the database should be as low as possible (10–20% or less). If a protein sequence for which the secondary structure is being predicted is too similar to any of the sequences in the database, then the prediction is usually better.

Because the performance of various prediction programs depends on the databases used and on the set of sequences for which the prediction is done, the comparison of accuracy of prediction of the various methods should be done very carefully by using cross-validation techniques. Recently, Cuff and Barton[30,31] proposed new databases of non-redundant domains for the unbiased testing of various prediction algorithms. The first database contained 396 sequences and the latest database contains 513 non-redundant sequences. We have used this database of 513 sequences in the present work.

The GOR Method of Protein Secondary Structure Prediction

The GOR program is one of the first major methods proposed for protein secondary structure prediction from sequence. The original article (GOR I) was published by Garnier, Osguthorpe, and Robson[14] in 1978, with the first letters of the authors' names forming the name of the program. The method has been continuously improved and modified during the last 20 years.[15–17,44] The first version (GOR I) used a rather small database of 26 proteins with about 4,500 residues. The next version (GOR II) used the enlarged database of 75 proteins of Kabsch and Sander containing 12,757 residues.[16] Both versions predicted four conformations (H, E, C, and turns T) and were using singlet frequency information within the window (so called: directional information). Starting with GOR III,[15] the number of predicted conformations was reduced to three (H, E, and C). The GOR III method started to additionally use information about the frequencies of pairs (doublets) of residues within the window, based on the same database as the earlier version. The latest version of the program is available online at the web based protein secondary structure prediction server (http://abs.cit.nih.gov/gor/) and is named GOR IV.[17] It uses 267 protein chains containing 63,566 residues. The GOR algorithm is based on the information theory combined with the Bayesian statistics. One of the basic mathematical tools of the information theory is the information function I(S,R):

$$I(S; R) = \log[P(S|R)/P(S)] \quad (7)$$

For the problem of the protein secondary structure prediction, the information function is defined as the logarithm of the ratio of the conditional probability P(S|R) of observing conformation S, [where S is one of the three states: helix (H), extended (E), or coil (C)] for residue R (where R is one of the 20 possible amino acids) and the probability P(S) of the occurrence of conformation S. The last publicly available version (GOR IV) of the program is using a database of 267 sequences with known secondary structure to calculate the information function I(S;R).

The conformational state of a given residue in the sequence depends not only on the type of the amino acid R but also on the neighboring residues along the chain within the sliding window. GOR IV uses a window of 17 residues, that is, for a given residue, eight nearest neighboring residues on each side are analyzed.

According to information theory, the information function of a complex event can be decomposed into the sum of information of simpler events, generally:

$$I(\Delta S; R_1, R_2, \ldots R_n) = I(\Delta S; R_1) + I(\Delta S; R_2|R_1) + I(\Delta S; R_3|R_1, R_2) + \cdots + I(\Delta S, R_n|R_1, R_2, \ldots R_{n-1}) \quad (8)$$

where the information difference is defined as:

$$I(\Delta S; R_1, R_2, \ldots R_n) = I(S; R_1, R_2, \ldots R_n) - I(n\text{-}S; R_1, R_2, \ldots R_n) \quad (9)$$

Here, n-S denotes all conformations different than S, that is, if S is H then n-S is E and C.

The GOR IV method assumes also that the information function is a sum of information from single residues (singlets) and pairs of residues (doublets) within the window of width 2d +1 (i.e., d = 8, for the window of 17 residues):

$$\log \frac{P(S_j, \text{LocSeq})}{P(n\text{-}S_j, \text{LocSeq})} = \frac{2}{2d+1} \sum_{n,m=-d}^{d} \log \frac{P(S_j, R_{j+m}, R_{j+n})}{P(n\text{-}S_j, R_{j+m}, R_{j+n})} - \frac{2d-1}{2d+1} \sum_{m=-\mathrm{d}}^{\mathrm{d}} \log \frac{P(S_j, R_{j+m})}{P(n\text{-}S_j, R_{j+m})} \quad (10)$$

Here the first summation is over doublets and the second summation is over singlets within the window centered around the j-th residue. The pair frequencies of residues R_j and R_{j+m} with R_j occurring in conformations S_j and n-S_j are calculated from the database. All 267 proteins in the GOR IV database have well-determined structures (with crystallographic resolution at least 2.5 Å). Using the frequencies calculated from the databases, the program can predict probabilities of conformational states for a new sequence.

The accuracy of the prediction with the GOR IV program based on single sequences (without multiple alignments) tested on the database of 267 sequences with the rigorous jack-knife methodology was 64.4%. Other methods (using single sequences) for the secondary structure predictions such as neural network methods or nearest-neighbor methods have similar or lower success rates.[30] A big advantage of the GOR method over other methods is that it clearly identifies all factors that are included in the analysis and calculates probabilities of all three conformational states. Because the GOR IV algorithm is computationally fast, it is possible to perform the full jack-knife procedure: each time when the prediction for the given sequence (of 267 sequences) is done, the sequence is removed from the database and the spectrum of frequencies used for the prediction is recalculated without including the information about the query sequence.

In a recent publication, Kloczkowski et al.[44] applied the evolutionary information for the secondary structure prediction by combining of the multiple sequence alignment for a small set of 12 proteins with the GOR IV algorithm. All 12 proteins in the set were chosen to represent various classes of folds. The proteins had well-determined resolution (better than 2.5 Å, mostly better than 2 Å) and had identity less than 20% to any of the 267 proteins in the database. We first performed binary alignments for each protein chain (from the set of 12 chains under study) with sequences in the PIR database by using the FASTA program. Then sequences having at least 20–30% pairwise identity with the target sequence were aligned by using the CLUSTAL program for the multiple alignment. Sequence comparison is an extremely important technique in computational biology. By comparing different sequences to each other, one can determine which parts of sequences are similar. The similarity of sequences is usually related to their evolutionary dependence and quality of the alignment of two sequences can be measured by a properly defined score. The GOR program takes sequence (in the FASTA format) and predicts its secondary structure. For each residue i along the sequence, the program calculates the probabilities p_H, p_E, and p_C and the secondary structure prediction (H, E, or C). The probabilities are normalized between 0 and 1 with:

$$p_H + p_E + p_C = 1 \quad (11)$$

Usually, the predicted conformational state corresponds to that with the highest probability, but sometimes the program makes exceptions to this rule.

We have previously applied the GOR algorithm to the multiple sequence alignments (for the 12 proteins) obtained by using CLUSTAL. The gaps in the alignments were skipped by the GOR algorithm during the calculation of probabilities p_H, p_E, and p_C for each residue in the multiple alignment but the information about position of gaps was retained. If we consider the multiple alignment as a matrix of size n × m, where m is the length of the alignment and n is the number of alignments, then each element of the alignment matrix is one of the 20 amino acid letters, or a gap (-) represented by the 21-st letter character. [The possibility of unknown residues (X), or nonstandard residues (B, Z) is also taken into consideration, and such residues are similarly treated as gaps by the GOR program.] The results of the calculations of the probabilities of the three secondary states p_H, p_E, and p_C for all residues in the alignments by the GOR program are stored as three matrices $P_H(i, j)$, $P_E(i, j)$, and $P_C(i, j)$ each of size n × m, such that $P_H(i, j)$ represents the probability p_H of the helical conformation for the j-th residue in the i-th multiple alignment sequence (with similar definition for the E and C conformational states).

If the (i,j)-th element of the alignment matrix is a gap (-) then the (i,j)-th elements of $P_H(i, j)$, $P_E(i, j)$, and $P_C(i, j)$ are set to zero. The gaps are neglected in the prediction of the secondary structure by the GOR algorithm but the information about them is retained in the alignment matrix and in the matrices $P_H(i, j)$, $P_E(i, j)$, and $P_C(i, j)$. In the final step, we calculate the average values (over all alignments) $\langle P_H(j)\rangle$, $\langle P_E(j)\rangle$, and $\langle P_C(j)\rangle$ of the elements of matrices $P_H(i, j)$, $P_E(i, j)$, and $P_C(i, j)$ at the j-th column in the alignment matrix A. We sum the probabilities $P_H(i, j)$ [and similarly $P_E(i, j)$ and $P_C(i, j)$] over all alignments ($0 \le i \le n$) at the j-th position in the sequence of multiple alignments, and divide this sum by the number of alignments containing non-gap entries at the j-th position. We skip columns that contain gaps at the j-th position in the query sequence. Then for each position j at the alignment sequence we calculate the maximum of the three numbers $\max\{\langle P_H(j)\rangle, \langle P_E(j)\rangle, \langle P_C(j)\rangle\}$. If, for example, the maximum corresponds to $\langle P_H(j)\rangle$ then the j-th position in the multiple sequence alignment is assigned the helical conformation with prob-

ability $\langle P_H(j) \rangle$. In this way, each position in the multiple sequence alignment is assigned a secondary structure.

This method proved to be successful in the prediction of the secondary structure, the accuracy of the prediction for the set of 12 proteins being 71.9%. By applying a correction technique to remove very short helices or sheets, and imposing the requirement that the conformations H and E must be separated by at least one C state the accuracy of the prediction increased to 74.4%. The set of 12 proteins is however too small for definitive estimation of the accuracy of the prediction, even though the proteins in the set were carefully chosen to have no identity to sequences in the database and to be a good unbiased representation of a larger population of proteins.

The main aim of this article was the systematic study of the prediction of the secondary structure by the GOR method using multiple alignments.

The Method

We made several changes to the GOR IV program to improve the accuracy of the secondary structure prediction even from a single sequence without the multiple alignments, and to ensure that the accuracy of the predictions satisfy the jack-knife cross-validation standards. All these modifications provide a basis for the new version of the program that we call GOR V. The new GOR V version of the program will soon be available on a web-based secondary structure Internet prediction server, replacing the current GOR IV version.

All modifications and improvements of the original GOR IV program incorporated into the GOR V version are listed below:

1. As mentioned in the introductory part of this article, we have enlarged the database of sequences with known secondary structure. The previous GOR database of 267 sequences was replaced by a new database of 513 non-redundant domains proposed by Cuff and Barton.[30,31] The database of 513 non-redundant domains containing 84,107 residues was downloaded from Barton's web site.[30,31] The strict application of the full jack-knife method for this database enables a highly accurate, objective, and unbiased calculation of the accuracy of the prediction, and an easy comparison with results of other prediction algorithms that use the non-redundant sequences as a prediction methodology standard.
2. We have optimized the parameters in the GOR algorithm to increase the accuracy of the prediction. The most important modification was the introduction of the decision constants in the final prediction of the conformational state. The GOR IV program had a tendency to over-predict the coil state (C) at the cost of the helical conformation (H), and to an even greatest extent at the cost of β-strands (E). We have therefore introduced decision parameters.

   The predicted probability of the coil (C) conformation must be greater by some critical margins than probability of either the (H) or (E) states to accept C as the winning conformation in the prediction process. The margin for the β-strands is greater than for helices, because strands were more often mis-predicted by the program as coils. The introduction of the decision constants significantly improves the predicted results by about 1.6%.
3. We have modified the GOR algorithm to include the triplet statistics within the window. The previous versions of the program used only single residue statistics (GOR I–II) or the combination of the single residue and pair residue statistics within the window (GOR III–IV). Now the GOR algorithm calculates statistics of singlets, pairs, and triplets for the secondary structure prediction. By introducing additional triplet statistics, the problem of optimization of the prediction becomes more complicated, and in the present version the triplets are only treated as a small perturbation to the optimization solution found for singlets with doublets. Because of this, the addition of the triplets improved the accuracy of the prediction by only 0.3%. It is probably possible to find a better optimization and obtain a further improvement to the prediction in the future, notably by using a larger database.
4. We have applied a resizable window for the GOR program. The previous version of the program (GOR IV) was using the window having a fixed width of 17 residues, that is, with eight residues on both sides of the central one. We have studied in detail the dependence of the accuracy of the prediction of the GOR algorithm on the size of the window. It is found that the accuracy of the prediction is slightly better for the smaller window of the width of 13 residues. The use of a smaller window has computational advantage in that it requires less computer memory and in turn permits us to include the triplet statistics (within the window) for the prediction of secondary structure. Because the number of triplets within the window of size N is N(N-1)(N-2)/6, the difference between the window of size 17 (680 triplets) and the new one of size 13 (286 triplets) is substantial. The Cuff and Barton[30,31] database on non-redundant sequences of protein domains also includes a significant number of short sequences, that are not domains, with many of them as short as 20–30 residues. Of course, the prediction of the secondary structure for such short sequences is very inaccurate, because of the artificial end effect of the window. (For residues at the beginning or at the end of the sequence, there are no neighbors on the left or on the left side within the window to provide proper statistics.) All window-based prediction programs have this problem and usually short sequences are omitted in the prediction and removed from the database. We have found that we can overcome this problem by using smaller windows for the prediction of the secondary structure of short sequences. The use of a window size of seven or nine residues gives surprisingly good (better than the average prediction) results for residues as short as 20–30 residues. We have therefore modified the program in such way that, depending on the length of the

query sequence, the GOR algorithm adjusts the width of the window used for the prediction. For sequences 25 residues or shorter, we use the window size of seven residues, for sequences longer than 25 but shorter than 51 residues, the window is nine residues, for sequences at least 51 residues long but shorter than 100 residues, the window is 11 residues, and for all sequences at least 100 residues long, the window is 13 residues. The introduction of the resizable window allowed us to include all 513 non-redundant sequences in the prediction procedure.

5. We have used the multiple sequence alignment for the secondary structure prediction. Instead of using first the FASTA pair alignments and then CLUSTAL multiple sequence alignments—the method applied in our previous study of the set of 12 protein chains—we used directly the multiple sequence alignments from the PSI-BLAST program for each of the 513 non-redundant sequences from the database. We run PSI-BLAST program using the nr database which contains all known databases: all non-redundant GeneBank CDS translations + PDB + SwissProt + PIR + PRF. We set the maximum number of iterations in the BLAST computations to five with an E value of 5×10^{-4}. This means that if in five iterations the BLAST alignment procedure is unable to converge, we use the partially converged alignments from the fifth iteration. Only in a few (four) cases (mainly for very short sequences) was the BLAST program unable to find any alignments ("no hits found" result) and in those cases the original single sequence was used for the prediction. The number of alignments varied considerably depending on the sequence. For some sequences, the BLAST program produced more than 2,000 sequences, whereas for some other sequences, only a few alignments. We first used all alignments generated by the BLAST algorithm and then we tried to select the range of the identity of the alignments with the query sequence that gave the best accuracy of the prediction. He have found that a small improvement in the prediction is obtained by removing the alignments that are too similar to the query sequence. We have tried various sequence identity thresholds and found that the best results are obtained by skipping all alignments that have identity greater than 97% to the query sequence. The effect of cutting off the alignments that are too similar to the query sequence is relatively strong, probably because the BLAST program produces a large number of such alignments. However, cutting off the alignments with low identity to the query sequence did not improve the prediction results, because the number of such alignments is small, so we include even very dissimilar alignments. Besides the identity threshold, we tried to use various methods of weighting of the alignments in the calculation of the accuracy of the prediction. Various weighting schemes related to the identity of the alignments with the query sequence gave almost similar predictions, so we treated all alignments similarly, except those with similarity more than 97% which were rejected.

The methodological procedure was the same as in our previous work[44] for the set of 12 protein chains, based on the calculation of the matrices of the probabilities of various (H, E, and C) secondary structure elements $P_H(i, j)$, $P_E(i, j)$, and $P_C(i, j)$ for each j-th residue in the i-th alignment (with the inclusion of alignment gaps). The gaps were skipped by the GOR program in the calculation of the probabilities of various secondary structure conformations, but the information about them was retained for averaging purposes. Then we calculated the averages over alignments $\langle P_H(j)\rangle$, $\langle P_E(j)\rangle$, and $\langle P_C(j)\rangle$ at the j-th position in the alignment by summing $P_H(i, j)$ [and similarly $P_E(i, j)$ and $P_C(i, j)$] over i, by dividing this sum by the number of alignments, excluding (in the alignment count) alignments with gaps at the j-th position. We have also skipped in the alignment matrix columns containing gaps in the query sequence, contracting the size of the matrix to the original length of the query sequence. The prediction of the secondary structure conformation for the j-th residue was based of the set of three probabilities $\{\langle P_H(j)\rangle, \langle P_E(j)\rangle, \langle P_C(j)\rangle\}$. In our previous study of the secondary structure prediction using the multiple sequence alignments for a set of 12 protein chains, the secondary structure of the j-th residue was assigned to the conformation with the largest probability value $\max\{\langle P_H(j)\rangle, \langle P_E(j)\rangle, \langle P_C(j)\rangle\}$. We have modified this assignment procedure by introducing decision constants, as described above in this section. The original GOR IV program over-predicted the coil (C) state, instead of the (H) or (E) state when the calculated probability of the coil state was slightly larger than the probabilities of (H) or (E). We have therefore introduced the decision constant thresholds. The coil state is now being predicted only if the calculated probability of the coil conformation is greater then the probability of the other states (H, E) plus the imposed thresholds (0.15 for E and 0.075 for H). The value of the threshold for the β-sheets is larger than for α-helices, because strands were more often erroneously predicted as coils.

We have performed all calculations for the translation of the eight-state DSSP assignments into the three secondary structure states H, E, and C the same as that used by the Frishman and Argos.[5] This means that DSSP states H and E were translated to H and E in the three-state code, and all other letters of the DSSP code were translated to coil (C). Additionally, similar to Frishman and Argos, we treated helices shorter than five residues (HHHH or less) and sheets shorter than three residues (EE or E) like coils. The main reason behind the application of the Frishman and Argos DSSP translation was that the GOR algorithm has a built-in correction scheme, which removes secondary structure segments that are too short (helices shorter than four residues, and sheets shorter than three residues), treating them as the most likely prediction errors. The Frishman and Argos assignment scheme is therefore highly compatible with the GOR program performance. It is however known that the Frishman and Argos transla-

TABLE I. Global Results for Secondary Structure Prediction, the Accuracy Matrix, and Parameters Q (for Each State) and Q_3 Per Residue

	Predicted			
Observed	H	E	C	Total
H	18,376	849	5,616	24,841
E	1,788	8,526	6,526	16,840
C	4,759	2,812	34,855	42,426
Total	24,923	12,187	46,997	84,107
Q_{pred}	73.7	70.0	74.2	
Q_{obs}	74.0	50.6	82.1	
Q_3		73.4		

TABLE II. The Parameter Q_3, Segment Overlap (Sov), J_1^{score} and J_2^{score} Averaged Over 513 Sequences in the Database and Their Corresponding Root-Mean-Square Deviations (σ)

	Average over 513 sequences	σ
Q_3	73.5	9.8
Sov	70.8	14.4
J_1^{score}	78.4	12.1
J_2^{score}	79.5	12.5

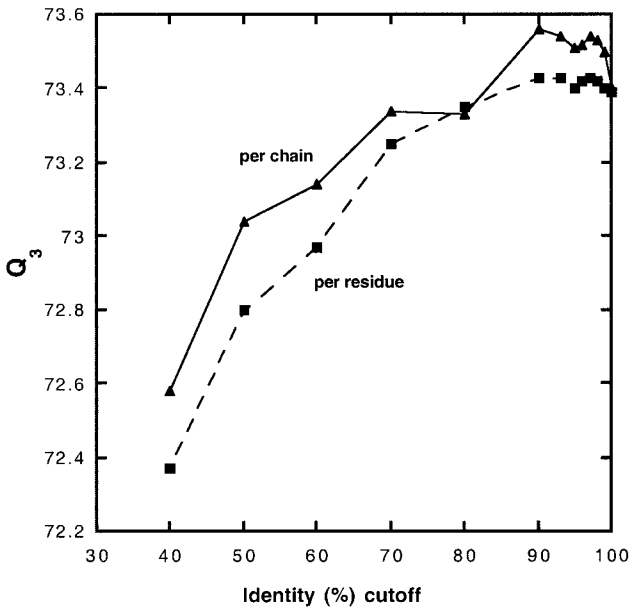

Fig. 1. Dependence of the accuracy of the predictions on the upper sequence identity (to the query sequence) cutoff threshold for sequences from the multiple-sequence alignment. The accuracy of prediction improves when we remove alignments more similar than 95–97% to the query sequence, but removing all sequences with identity more than 90% lowers the accuracy of the predictions.

tion of the secondary structure gives slightly higher accuracies of prediction, than the strict translation that maintains short helices and strands.

RESULTS

The results obtained are shown in two tables. Table I shows all elements of the accuracy matrix A and the parameter Q_3 [defined by Eq. (3)] and two kinds of parameters Q for the individual correctness of prediction for each secondary structure class (H, E, and C), Q_{pred} and Q_{obs}. The parameter Q_{obs} is defined by Eq. (5), whereas Q_{pred} has similar functional form [such as Eq. (5)], with the N_i [defined by Eq. (2)] replaced by n_i [from Eq. (1)]. The analysis of the results in Table I shows that the GOR V program is most successful in the prediction of coils, quite good in the prediction of helices, but less successful in specifying of β-sheets. The average value $Q_3 = 73.4$ for the GOR V algorithm with multiple alignments is much better than the accuracy of the GOR IV method based on single sequences Q_3, which was around 65%. Cuff and Barton[30] cross-validated the accuracy of the GOR IV algorithm using their earlier database of 396 non-redundant sequences and a similar translation of the eight-letter DSSP assignment code into the three-letter secondary structure code. They reported 64.6% accuracy (see Table XIII in Ref.[30]). The accuracy of the GOR V algorithm based on single sequences is $Q_3 = 66.9$ (per residue) and 67.5 per chain. This represents more than a 2% increase attributable to various optimizations applied in the GOR V version.

Table I shows the accuracy of prediction per residue. Another way of presenting the results is to calculate averages over the total number of chains in the database. The advantage of this method is the possibility of calculation of statistical mean square deviations (σ^2) of the averaged quantities. The results of these calculations are shown in Table II, where the average values are shown together with the root-mean-square deviation (σ).

We have calculated the following quantities for Table II:

1. The Q_3 prediction value per chain (averaged over 513 chains) and its root-mean-square deviation.
2. The average segment overlap Sov (as recently redefined by Zemla et al.[9]) and the root-mean-square deviation of the segment overlap Sov.
3. The average segment overlap J_1^{score} and its root-mean-square deviation.
4. The average segment overlap J_2^{score} and its root-mean-square deviation.

The segment overlap Sov is defined in reference[9], and the two new segment overlap measuring scores J_1^{score} and J_2^{score} will be defined later in this section. The accuracy of prediction measured by Q_3 based on the Frishman and Argos translation of the DSSP assignments to the three secondary structure states is 73.5% per chain for all 513 non-redundant sequences. The average per chain is slightly higher than per residue (73.4) because the performance of the GOR program is better for short- and middle-sized length chains in the database, than for very long ones. It should be noted that Frishman and Argos[5] reported an accuracy of 74.8% (75% in the title of their publication), by using the PREDATOR algorithm and the 125 sequences from the Rost and Sander[18] database. The cross-validation of the performance of various prediction algorithms on the database of the 396 non-redundant sequences developed

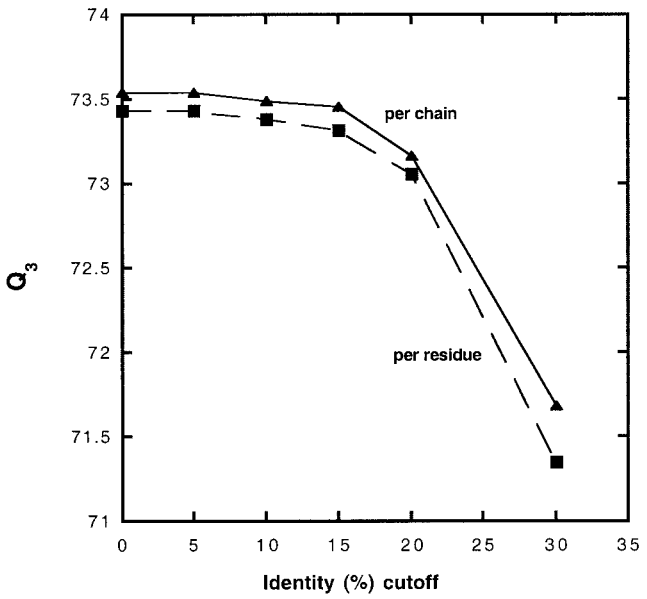

Fig. 2. Dependence of the accuracy of the predictions on the lower sequence identity (to the query sequence) cutoff threshold for sequences from the multiple sequence alignment. The removal of the sequences with very low identity to the query sequence does not improve the accuracy of the predictions, but neither does including them reduce the quality of the predictions.

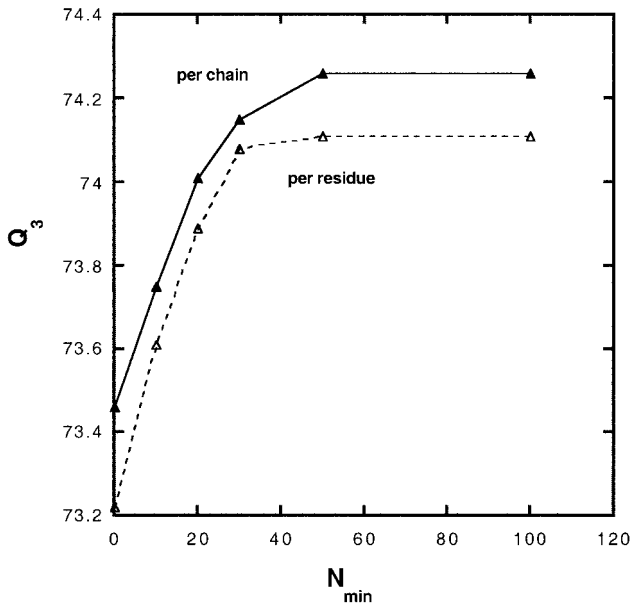

Fig. 3. The accuracy of the prediction as a function of the minimum number of PSI-BLAST alignments N_{min} used for the prediction. The best predictions are obtained for sequences having at least 50 alignments.

by Cuff and Barton[30] has demonstrated drops in the prediction levels (in comparison to the Rost and Sander database) from 1.3% to 2.7%, depending on the prediction program. The application of the full jack-knife methodology additionally lowers the reported accuracies.

We have found that the accuracy of the prediction actually increases if we remove from the multiple sequence alignment those chains that are almost identical to the query sequence. This is illustrated in Figure 1 where the accuracy of the prediction is plotted against the values of an upper identity percentage cutoff. The upper identity cutoff means that all sequences with identity (to the query sequence) higher than this upper limit are removed during the GOR prediction process. Figure 1 shows the interesting result that the removal of sequences with an identity larger than about 95–97% improves the prediction, whereas removing sequences that have identity lower than 90% to the query sequence lowers the accuracy of the prediction. We have also studied the opposite effect of a lower identity cutoff that removes from the multiple sequence alignment chains that have very low identity to the query sequence. The results are shown in Figure 2 where the accuracy of the prediction is plotted against the lower identity cutoff. We have found that the removal of the most dissimilar sequences does not influence the accuracy of the prediction, because of the small number of such sequences. These results show that all PSI-BLAST alignments (except the most identical ones) should be included, if possible, in the secondary structure prediction process. The neglect of the alignments with low identity to the query sequence will not help, but to the contrary may hinder the accuracy of the prediction. We have also tried various schemes of weighting the multiple sequence alignments depending on their identity to the query sequence in the process of the calculation of the averages over alignments $\langle P_H(j)\rangle$, $\langle P_E(j)\rangle$, and $\langle P_C(j)\rangle$ (for the j-th position in the alignment). We have found the accuracy of the predictions is almost the same for various weighting schemes, except for removing the sequences with highest similarity to the query sequence.

The accuracy of the GOR prediction is additionally increased if we limit prediction to chains having a sufficiently large number of alignments. We have tried to run the GOR program for various alignment number thresholds. The results are shown in Figure 3 where the accuracy of the prediction is plotted as a function of the minimum number of alignments N_{min} required for the query sequence. This means that all sequences from the Cuff and Burton database of 513 non-redundant domains, which have less than N_{min} PSI-BLAST produced alignments are skipped in the process of the calculation of the accuracy of prediction. Of course, the increase in the N_{min} threshold limit decreases the number of sequences (from the set of 513) that satisfy this limit. The value $N_{min} = 0$ corresponds to the case when all 513 sequences (even those that have no PSI-BLAST alignments) are used to calculate the accuracy of the prediction. The best prediction results were obtained for sequences having at least 50 PSI-BLAST alignments. There are 375 chains among the 513 domains in the database for which the PSI-BLAST program finds at least 50 alignments. The value of the Q_3 coefficient calculated for these 375 chains has increased to 74.3% (per chain) and 74.2% per residue.

It should be noted that all these results are very close to the accuracies obtained by using GOR V algorithm without the jack-knife procedure. The calculated value of Q_3 for 513 sequences without the jack-knife method was 74.9%

TABLE III. Observed Sequence and Four Different Predictions With Weights Illustrating the Calculation of Coefficients J_1^{score} and J_1^{score} for Prediction 1

Observed	CCHHHHHHHHHCCC	Sov	J_1^{score}	J_2^{score}
Prediction 1	CCCHHHCHHHHCCC	65.2	80.6	77.8
Weights for calculation of J_1^{score} for Prediction 1	11123454321121			
Weights for calculation of J_2^{score} for Prediction 1	11124898421121			
Prediction 2	CCCCHHHHHCCCCC	77.9	80.6	86.7
Prediction 3	CCHHHHCCCCCCCC	67.9	51.6	46.7
Prediction 4	CCCCCHHHHCCCCC	68.6	71.0	77.8

(per chain) and 74.7% (per residue). For 375 sequences having at least 50 alignments, the Q_3 calculated without the jack-knife procedure was 75.2% (per chain) and 75.1% per residue. The difference between the prediction based on multiple sequence alignments obtained with the full jack-knife and without the jack-knife procedure is less than 1.5%. This is much less than the prediction for single sequences (without multiple alignment) where the imposition of the jack-knife requirement leads to a significant decrease of about 5% in the prediction accuracy.

Besides Q_3, we have calculated the segment overlaps (Sov) using the recent re-definition of Sov by Zemla et al.[9] This new definition of Sov ensures that the segment overlap is properly normalized ($0 \leq \text{Sov} \leq 100$). The old definition of Sov sometimes gave values of Sov larger than 100. The calculated values of Sov was 70.8 and its root-mean-square deviation [σ(Sov)] was 14.4. These results are shown in the second row in Table II.

We think, however, that the segment overlap Sov does not properly measure all aspects of the quality of overlaps. The Sov puts a great weight on the requirement that the predicted segments should not be disjoint. Every disruption in the continuity of the predicted sequence leads to large penalties. This is illustrated in Table III showing the observed sequence and four different predictions (Prediction 1–4).

The third column in Table III shows Sov calculated for each prediction. Sov for Prediction 1 is 65.2% and for Prediction 2 is 77.9%, only because the Prediction 1 contains the one C residue disrupting the helical sequence. It does not matter that the first prediction has the correct prediction of seven of the nine residues in the H state, whereas the second one has only five of the nine H residues. Additionally, the Sov values reflect little about the relative positions of overlapping segments as illustrated by comparing the observed sequence with Predictions 3 and 4 in Table III. The Sov for Predictions 3 and 4 are almost the same: 67.9% and 68.6%, respectively. Prediction 4 is, in our opinion, better than Prediction 3 because centers of helices for the observed and predicted sequence are close, and should be given a much better overlapping score.

To cope with these problems, we introduce two new coefficients J_1^{score} and J_2^{score} measuring the quality of segment overlaps (Di Francesco et al., unpublished results). Both coefficients are defined similarly by Eq. (12) with the only difference being in the definition of weights W_i.

$$J^{score} = \frac{\sum_{i=1}^{n} W_i \delta_{ij}}{\sum_{i=1}^{n} W_i} 100 \qquad (12)$$

Each residue in secondary structure segment is given a weight, depending on the position of the residue in the segment. The residues on both ends of a secondary structure segment are given weights of 1, next residues are given weights 2, 3, 4…, growing in the algebraic series form (1, 2, 3, 4, 5, 6…). While calculating the J_1^{score}, we sum up the weights for residues having the same secondary structure in the observed and predicted sequence [corresponding to the Kronecker δ_{ij} in Eq. (12), which is one, if residue i in the observed sequence and j in the prediction are in the same state and at the same position, i = j, otherwise δ_{ij} is zero] and divide this sum by the sum of all weights in the observed sequence of n residues. The illustration of this method is given in Table III by showing weights for calculation of these coefficients for Prediction 1. The values of J_1^{score} for each prediction is given in the fourth column of Table III. For the J_2^{score}, the weights are similarly defined, except that instead of growing in the algebraic series form (1, 2, 3, 4, 5, 6…) from each end of the segments, they have the geometric series form for the first four members, and then they grow like the J_1^{score}, i.e., they form a series (1, 2, 4, 8, 9, 10, 11…) for the each end of a segment, as shown in Table III for Prediction 1.

The calculated values of J_2^{score} for each prediction are shown in the last column of Table III. It should be noted that the definition of J_1^{score} and J_2^{score} is much simpler than the definition Sov and these coefficients account for the problems mentioned above. The calculated average values of coefficients J_1^{score} and J_2^{score} for the database of 513 sequences are shown in Table II together with their corresponding root-mean-square deviations. The J_2^{score} (79.5) is slightly higher than the J_1^{score} (78.4).

Single average coefficients are not, however, a good measure of the quality of the secondary structure prediction. Much more information is contained in the distribution of segment lengths of various secondary structure elements. Figure 4 shows the observed distribution of lengths of helices in the database (solid line) and the similar distribution of lengths of helices for the GOR V predictions. We count the number N_L of segments in the database having length L. A similar count is done for the

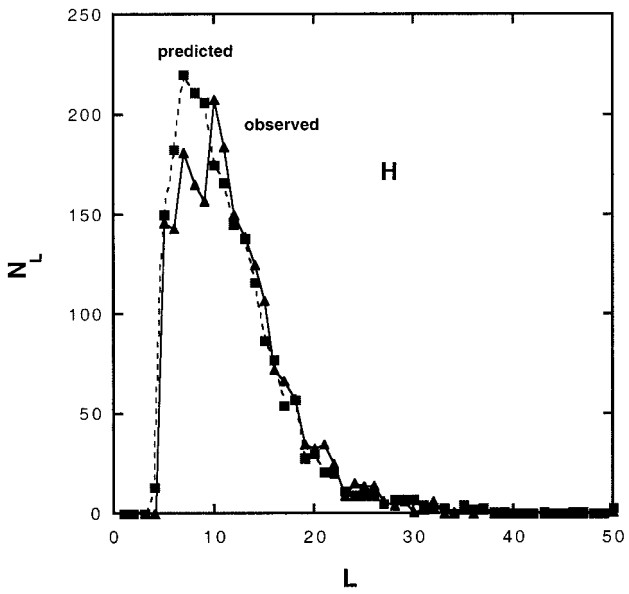

Fig. 4. The observed distribution N_L of the lengths L of H in the Cuff and Barton database of 513 non-redundant sequences (solid line) and the distribution predicted by the GOR V program (dashed line) for these same sequences.

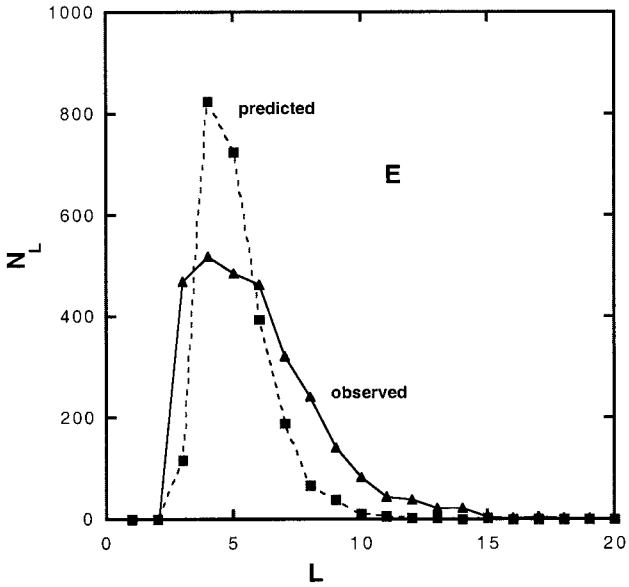

Fig. 5. The observed distribution N_L of the lengths L of β-sheets in the Cuff and Barton database of 513 non-redundant sequences (solid line) and the distribution predicted by the GOR V program (dashed line) for these same sequences.

segments from the secondary structure predictions. Figure 4 shows that the distribution of lengths of helices is rather well predicted by the GOR program. The main difference is the segment length at the maximum of the distribution, the observed value is L = 10, whereas the GOR program predicts the maximum at L = 7, but nonetheless the breadth of the predicted and observed distributions are extremely close to one another.

Figure 5 shows similar distributions for lengths L of β-sheets (E). Because sheets are usually much shorter than helices, the length L scale in Figure 5 differs from the distribution of Figure 4. Figure 5 shows that the GOR program is less successful in the prediction of strand lengths. Whereas the maximum of both distributions observed (solid line) and predicted (dashed line) coincide at L = 4, the shapes of the distributions in Figure 5 differ substantially. The predicted distribution is too narrow and highly over-predicts the number of segments with the length 4 and 5, and under-predicts the number of segments longer than 5. This interesting result indicates where, in the future, improvement to the GOR algorithm can be made, by implementing decision constants based on the length of the β-sheet segments.

DISCUSSION

We have shown that the GOR prediction algorithm based on information theory and incorporating multiple sequence alignment information is quite successful in its accuracy of secondary structure prediction. The calculated accuracy of the prediction was based on the Cuff and Barton database of 513 non-redundant domains (containing 84,107 residues) with a rigorous application of the jack-knife procedure. The prediction accuracy is near the accuracy that we found in our earlier work[44] on a set of 12 protein chains, which shows that that small set was surprisingly well chosen to represent accurately much larger groups of proteins.

The accuracy of the prediction with the GOR method seems to be 2–3% less than the published accuracies of prediction for the most successful prediction methods based on neural networks but the actual comparison of the accuracy of the prediction to other prediction methods should be done by the existing automatic evaluation of server predictions on the same series of proteins, such as project EVA[45] or LiveBench[46] (although the LiveBench project is mainly dedicated to the three-dimensional predictions).

The advantage of the GOR method in comparison to neural network based predictions is that all parameters of the algorithm are fully controllable, have direct physical meanings, and provide us with insights about the relation between sequence and the structure. The neural network methods work like "black boxes," providing no understandable relation between the input and the output, and therefore do not have the advantage of the present method.

The GOR method benefits from its relative simplicity and low computational resource requirements, which makes it possible to do predictions in real time without long waits for results. The GOR method predicts the probabilities of the three conformational states for each residue in the sequence, and this information can be used for further analyses or simulations. Most of the other prediction methods do not provide so much information, and usually the most they give is a confidence level of the prediction for a given position in the sequence. The probabilities of various secondary structure conformations give us direct information about the confidence level of the

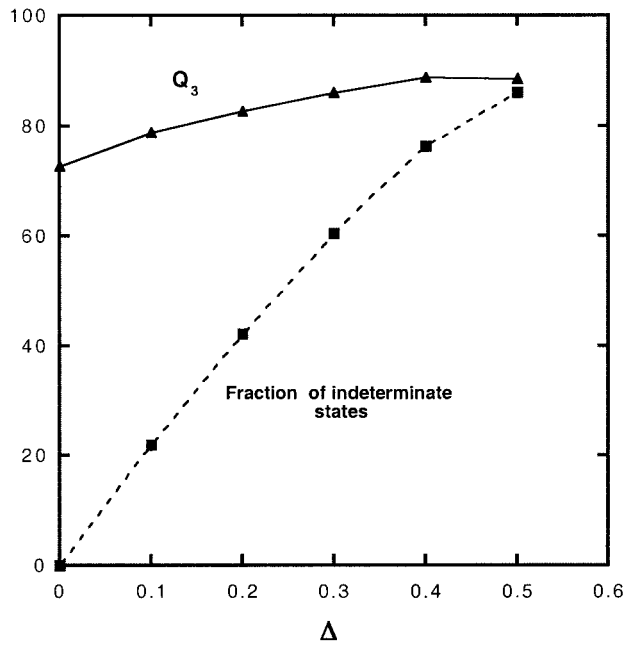

Fig. 6. The accuracy of the prediction Q_3 (solid line) and the corresponding percentage of unpredicted states (broken line) as a function of the parameter Δ ($0 < \Delta < 1$). We impose the requirement that the state with the highest probability predicted by the GOR program with multiple sequence alignment must be larger than the probability of the second-most probable state by a certain minimum value Δ. All probabilities are normalized in account with Eq. (11).

GOR prediction. We have initiated earlier studies on this problem with a set of 12 proteins with multiple sequence alignment information.[44] We analyzed the relation between the accuracy of the prediction and the strength of the prediction measured by the difference Δ ($0 < \Delta < 1$) between the probability of the predicted state (with the highest probability) and the probability of the next-most probable state. The probabilities of the GOR prediction are normalized to 1 [Eq. (11)]. Figure 6 shows the results of calculations with varying values of the parameter Δ. The solid line shows the accuracy of the prediction for strongly predicted residues (satisfying the threshold Δ requirement) as a function of the parameter Δ. If the threshold Δ is large enough ($\Delta \approx 0.4$), the prediction accuracy reaches almost the 89% accuracy level. The broken line shows the percentage of residues with indeterminate states (weakly predicted) that do not satisfy the Δ threshold requirement. The imposition of the Δ threshold increases significantly the accuracy of the prediction, but the percentage of residues predicted with high confidence level expectedly diminishes with the increase in Δ. We are planning in the future to incorporate the information about the residues predicted with high confidence into the GOR program as an input for the next level of prediction, built upon those firmly predicted residues. This approach may improve the performance of the GOR algorithm, because the secondary structure segments can be viewed to be built around those strongly nucleating residues.

The main failure of the GOR method is the very low accuracy (about 50%) in the prediction for β-sheets (E). In future work, we will try to combine the GOR method with the global composition-based prediction methods[47–51] to improve the accuracy of the prediction of β-sheets and hence the total accuracy of the prediction.

It is worth mentioning that because the GOR method is computationally simple, it allows use of the full jack-knife procedure and computation of the prediction in real time on a personal computer. The neural network based prediction methods require substantially larger computational resources during the network learning process, so that the jack-knife is almost always done by removing not a single sequence but a whole group of sequences from the database. The accuracy of the prediction for the new GOR V method with multiple sequence alignments is nearly (within the 2–3% accuracy limit) as good as neural network predictions. This demonstrates clearly that the GOR information theory based approach—a method with more than 20 years' history—is still viable and at the front line of secondary structure prediction methods.

REFERENCES

1. Kabsch W, Sander C. A dictionary of protein secondary structure. Biopolymers 1983;22:2577–2637.
2. The address of the FTP server for the DSSP assignments for chains from the PDB is ftp://ftp.embl-heidelberg.de/pub/databases/dssp.
3. Rost B, Sander C. Third generation prediction of secondary structure. In: Webster DM, editor. Protein structure prediction: methods and protocols. Totowa, NJ: Humana Press; 2000. p 71–95.
4. Moult J, Judson R, Fidelis K, Pedersen JT. A large scale experiment to assess protein structure prediction. Proteins 1995;23:I–IV.
5. Frishman D, Argos P. Seventy-five percent accuracy in protein secondary structure prediction. Proteins 1997;27:329–335.
6. Biou V, Gibrat JF, Levin J, Robson B, Garnier J. Secondary structure prediction: combination of three different methods. Protein Eng 1988;2:185–191.
7. Salamov AA, Solovyev VV. Prediction of protein secondary structure by combining nearest-neighbor algorithm and multiple sequence alignments. J Mol Biol 1995;247:11–15.
8. Matthews BB. Comparison of the predicted and observed secondary structure of T_4 phage lysozyme. Biochim Biophys Acta 1975; 405:442–451.
9. Zemla A, Venclovas C, Fidelis K, Rost B. A modified definition of Sov, a segment-based measure of protein secondary structure prediction assessment. Proteins 1999;34:220–223.
10. Rost B, Sander C, Schneider R. Redefining the goals of protein secondary structure prediction. J Mol Biol 1994;235:13–26.
11. Chou PY, Fasman GD. Prediction of protein conformation. Biochemistry 1974;13:211–215.
12. Lim VI. Structural principles of the globular organization of protein chains: a stereochemical theory of globular protein secondary structure. J Mol Biol 1974;88:857–872.
13. Lim VI. Algorithm for prediction of α-helical and β-structural regions in globular proteins. J Mol Biol 1974;88:873–894.
14. Garnier J, Osguthorpe DJ, Robson B. Analysis and implications of simple methods for predicting the secondary structure of globular proteins. J Mol Biol 1978;120:97–120.
15. Gibrat JF, Garnier J, Robson B. Further developments of protein secondary structure prediction using information theory: new parameters and consideration of residue pairs. J Mol Biol 1987;198: 425–443.
16. Garnier J, Robson B. The GOR method for predicting secondary structures in proteins. In: Fasman GD, editor. Prediction of protein structure and the principles of protein conformation. New York: Plenum Press; 1989. p 417–465.
17. Garnier J, Gibrat JF, Robson B. GOR method for predicting protein secondary structure from amino acid sequence. Methods Enzymol 1996;266:540–553.

18. Rost B, Sander C. Prediction of protein secondary structure at better than 70% accuracy. J Mol Biol 1993;232:584–599.
19. Rost B. Sander C, Schneider R. PHD: an automatic mail server for protein secondary structure prediction. Comput Appl Biosci 1994; 10:53–60.
20. Holley LH, Karplus M. Protein secondary structure prediction with a neural network. Proc Natl Acad Sci USA 1989;86:152–156.
21. Qian N, Sejnowski TJ. Predicting the secondary structure of a globular proteins using neural network models. J Mol Biol 1989; 202:865–884.
22. Stolorz P, Lapedes A, Xia Y. Predicting protein secondary structure using neural net and statistical methods. J Mol Biol 1992;225: 1049–1063.
23. Petersen TN, Lundegaard C, Nielsen M, et al. Prediction of protein secondary structure at 80% accuracy. Proteins 2000;41:17–20.
24. Jones TD. Protein secondary structure prediction based on position specific scoring matrices. J Mol Biol 1999;292:195–202.
25. Levin JM, Robson B, Garnier J. An algorithm for secondary structure determination in proteins based on sequence similarity. FEBS Lett 1986;205:303–308.
26. Levin JM, Garnier J. Improvements in secondary structure prediction method based on a search for local sequence homologies and its use as a model building tool. Biochim Biophys Acta 1988;955: 283–295.
27. Salamov AA, Solovyev VV. Protein secondary structure prediction using local alignments. J Mol Biol 1997;268:31–36.
28. Yi T-M, Lander ES. Protein secondary structure prediction using nearest-neighbor methods. J Mol Biol 1993;232:1117–1129.
29. Salzberg S, Cost S. Predicting protein secondary structure with nearest-neighbors algorithm. J Mol Biol 1992;227:371–374.
30. Cuff JA, Barton GJ. Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. Proteins 1999;34:508–519.
31. Cuff JA, Barton GJ. Application of multiple sequence alignment profiles to improve protein secondary structure prediction. Proteins 2000;40:502–511.
32. King RD, Sternberg MJ. Machine learning approach for prediction of protein secondary structure. J Mol Biol 1990;216:441–457.
33. Barton GJ. Protein secondary structure prediction. Curr Opin Struct Biol 1995;5:372–376.
34. Karplus K, Barrett C, Hughley R. Hidden Markov models for detecting remote protein homologies. Bioinformatics 1998;14:846–856.
35. Ouali M, King RD. Cascaded multiple classifiers for secondary structure prediction. Protein Sci 2000;9:1162–1176.
36. Rost B. Review: protein secondary structure prediction continues to rise. J Struct Biol 2001;134:204–218.
37. Zvelebil MJ, Barton GJ, Taylor WR, Sternberg MJE. Prediction of protein secondary structure and active sites using alignment of homologous sequences. J Mol Biol 1987;195:957–961.
38. Levin JM, Pascarella S, Argos P, Garnier J. Quantification of secondary structure prediction improvement using distantly related proteins. Protein Eng 1993;6:849–854.
39. Rost B. PHD: predicting one-dimensional protein structure by profile based neural networks. Methods Enzymol 1996;266:525–539.
40. Altschul SF, Madden TL, Schaffer AA, et al. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 1997;25:3389–3402.
41. Di Francesco V, Garnier J, Munson PJ. Improving protein secondary structure prediction with aligned homologous sequences. Protein Sci 1996;5:106–113.
42. Russell RB, Barton GJ. The limits of protein secondary structure prediction accuracy from multiple sequence alignment. J Mol Biol 1993;234:951–957.
43. Lecompte O, Thompson JD, Plewniak F, Thierry JC, Poch O. Multiple alignment of complete sequences (MACS) in the postgenomic era. Gene 2001;270:17–30.
44. Kloczkowski A, Ting K-L, Jernigan RL, Garnier J. Protein secondary structure prediction based on the GOR algorithm incorporating multiple sequence alignment information. Polymer 2002;43: 441–449.
45. Rost B, Eyrich VA, Przybylski D, et al. EVA: evaluation of automatic protein structure prediction services. www.http://cubic.bioc.columbia.edu/eva.
46. Rychlewski L, Fischer D. LiveBench: continuous benchmarking of prediction servers. www.http://bioinfo.pl/LiveBench.
47. Zhang CT, Chou KC. An optimization approach to predicting protein structural class from amino acid composition. Protein Sci 1992;1:401–408.
48. Chou KC. Does the folding type of protein depend on its amino acid composition? FEBS Lett 1995;363:127–131.
49. Eisenhaber F, Imperiale F, Argos P, Frommel C. Prediction of secondary structural content of proteins from their amino acid composition alone. I. New analytic vector decomposition methods. Proteins 1996;25:157–168.
50. Eisenhaber F, Frommel C, Argos P. Prediction of secondary structural content of proteins from their amino acid composition alone. II. The paradox with secondary structural class. Proteins 1996;25:169–179.
51. Bahar I, Altigan AR, Jernigan R, Erman B. Understanding the recognition of protein structural classes by amino acid composition. Proteins 1997;29:172–185.