PROTEINS: Structure, Function, and Genetics 40:451–462 (2000)

Protein Fold Recognition by Total Alignment Probability

Jadwiga R. Bienkowska,[1]* Lihua Yu,[2] Sophia Zarakhovich,[1] Robert G. Rogers Jr.,[1] and Temple F. Smith[1]
[1]*BioMolecular Engineering Research Center, College of Engineering, Boston University, Boston, Massachusetts*
[2]*AstraZeneca, R&D Boston, Cambridge, Massachusetts*

ABSTRACT **We present a protein fold-recognition method that uses a comprehensive statistical interpretation of structural Hidden Markov Models (HMMs). The structure/fold recognition is done by summing the probabilities of all sequence-to-structure alignments. The optimal alignment can be defined as the most probable, but suboptimal alignments may have comparable probabilities. These suboptimal alignments can be interpreted as optimal alignments to the "other" structures from the ensemble or optimal alignments under minor fluctuations in the scoring function. Summing probabilities for all alignments gives a complete estimate of sequence-model compatibility. In the case of HMMs that produce a sequence, this reflects the fact that due to our indifference to exactly how the HMM produced the sequence, we should sum over all possibilities. We have built a set of structural HMMs for 188 protein structures and have compared two methods for identifying the structure compatible with a sequence: by the optimal alignment probability and by the total probability. Fold recognition by total probability was 40% more accurate than fold recognition by the optimal alignment probability. Proteins 2000;40:451–462.** © 2000 Wiley-Liss, Inc.

Key words: fold recognition; protein structure; Viterbi algorithm; suboptimal alignments; HMM

INTRODUCTION

Protein fold-recognition methods have evolved into viable tools that help to deduce protein structure and function.[1] The ultimate goal of fold recognition is to predict protein structure by identifying the correct fold (structural template) among already-solved protein structures or models and correctly aligning the protein sequence onto the structural model. Most fold-recognition methods use Boltzmann statistics to interpret probabilistic scoring functions.[2–12] Sequence-to-structure alignments are evaluated in terms of a scoring function and the score of the alignment is interpreted as a "free energy" of the sequence in the conformation imposed by the alignment. This interpretation dictates that the most probable sequence-to-structure alignment is the one with the lowest "free energy." If one assumes that most structural models used for fold recognition represent an ensemble of similar structures, then the most probable (or lowest free energy) alignment represents only one of the fold variants. Thus, we investigated a fold-recognition procedure that evaluates the sequence-model compatibility by using the sum of the probabilities of all sequence-to-structure alignments. The mathematical justification for such a method was discussed previously for HMMs[13] and in general terms for a threading approach to protein structure prediction.[14,15] From the probabilistic viewpoint, the first approach attempts to maximize P(seq | Model,optimal-alignment) as a function of the alignment, while the second approach sums over all alignments.

The optimal sequence-to-structure alignment is rarely the correct one.[2,16,18] This situation reflects two facts about mathematical models of protein structure and structure prediction by threading. First, scoring functions that are used to evaluate sequence-to-structure alignments are statistical approximations of the "true" scoring function or "free energy." In consequence, the set of suboptimal alignments should be seen as a set of optimal alignments under expected minor fluctuations in the scoring function. Second, a structure model should be seen as a statistical representation of an ensemble of similar structures or expected variations about a unique fold topology. A set of suboptimal alignments can be interpreted as optimal alignments to structural variants of the same fold. According to the theory of Hidden Markov Models (HMMs), summing probabilities for all sequence-to-structure-model alignments gives the rigorous probability of observing the sequence given the structure model P(seq | Model).[13]

In our approach, the structure is modeled as a Discrete State Model (DSM)[13,19], which is mathematically represented as an HMM. This is a linear representation of the 3D protein structure that is essentially equivalent to a set of structural profiles.[20] The distinction between a DSM and an HMM is that HMMs are traditionally trained by an automated analysis of historical data. DSMs, in contrast, are physically-based probabilistic models. DSMs are built by using physically motivated structural building blocks. Other Hidden Markov Models for protein structure prediction or fold recognition have been proposed recently.[21] These HMMs are constructed from a generic HMM module. Subsequently, the generic HMM is trained, using a set of protein sequences that adopt similar 3D structures, to represent a structural fold. The potential problem with trained HMMs is that only the type of fold variation seen in the training set will be encoded in the model. For example, if only two surface loops are observed to vary in length in the training set, then other surface loops will be

*Correspondence to: J.R. Bienkowska, BMERC, Boston University, 36 Cummington Street, Boston MA 02215. E-mail: jadwiga@darwin.bu.edu

Received 12 October 1999; Accepted 23 March 2000

TABLE I. Results of the Fold-Recognition Experiments for 188 SCOP Superfamily Representatives[†]

	Filtering			Viterbi				Filtering			Viterbi		
Sequence PDB code	Top family probability	Top family PDB code	Native family rank	Top family probability	Top family PDB code	Native family rank	Sequence PDB code	Top family probability	Top family PDB code	Native family rank	Top family probability	Top family PDB code	Native family rank
1531	0.83	1531	1	0.99	1531	1	1a17	0.94	1a17	1	0.99	1a17	1
1a1x	0.63	1cmcA	18	0.50	1nsgB	26	1a32	0.89	1a32	1	0.99	1a32	1
1a62	0.93	1a62	1	0.61	2mhr	4	1a68	0.77	1a68	1	0.44	1nsgB	7
1a6jA	0.98	1a6jA	1	0.62	1flp	2	1a9t	0.99	1a9t	1	0.99	1a9t	1
1aa7B	0.98	1aa7B	1	0.62	1aa7B	1	1aazA	0.72	1aazA	1	0.50	1cmcA	4
1ab8B	0.85	1ab8B	1	0.51	1nfn	2	1acx	0.37	1acx	1	0.42	11kkA	29
1add	0.99	1add	1	0.98	1add	1	1ae9B	0.98	1ae9B	1	0.64	1nfn	2
1aep	0.88	1aep	1	0.99	1aep	1	1aerB	0.99	1aerB	1	0.89	1aerB	1
1af5	0.37	1af5	1	0.81	256bA	10	1ahq	0.46	1ravA	7	0.37	1flp	8
1air	0.99	1air	1	0.77	1cem	4	1aj2	0.99	1aj2	1	0.54	1ribA	2
1ako	0.99	1ako	1	0.99	1ako	1	1alkA	0.99	1alkA	1	0.77	1ft1A	2
1am2	0.99	1am2	1	0.99	1am2	1	1amk	0.87	1gky	7	0.99	21bd	17
1amp	0.99	1amp	1	1.00	1amp	1	1amx	0.94	1amx	1	0.95	1amx	1
1an7A	0.95	1an7A	1	0.98	2asr	3	1aol	0.99	1aol	1	0.99	1aol	1
1apa	0.99	1apa	1	1.00	1apa	1	1av6A	0.99	1av6A	1	0.98	1av6A	1
1awd	0.99	1awd	1	0.96	1awd	1	1axn	0.99	1axn	1	1.00	1axn	1
1ay9B	0.76	1ay9B	1	0.45	1cewI	4	1ayi	0.98	1ayi	1	0.99	1ayi	1
1ba7A	0.37	1cghA	4	0.82	1nfn	65	1bam	0.92	1bam	1	0.73	1nfn	2
1bgc	0.99	1bgc	1	0.99	1bgc	1	1bkrA	0.91	1bkrA	1	0.70	1bkrA	1
1ble	0.65	1cyw	2	0.79	1aep	7	1bme	0.99	1bme	1	0.55	1cby	2
1btn	0.82	1btn	1	0.86	256bA	8	1bv1	0.99	1bv1	1	0.59	1flp	2
1c52	0.99	1c52	1	0.95	1c52	1	1cby	0.99	1cby	1	0.99	1cby	1
1cem	1.00	1cem	1	1.00	1cem	1	1cewI	0.97	1cewI	1	0.90	1cewI	1
1cex	0.99	1cex	1	0.99	1cex	1	1cghA	0.56	1cghA	1	0.80	1nfn	22
1cgmE	0.67	6fd1	18	0.60	2asr	5	1chd	0.99	1chd	1	0.88	1chd	1
1cmcA	0.60	1cmcA	1	0.76	256bA	3	1cpt	1.00	1cpt	1	0.99	1cpt	1
1cyw	0.99	1cyw	1	0.83	1nfn	2	1deaB	0.99	1deaB	1	0.99	1deaB	1
1dhpA	1.00	1dhpA	1	1.00	1dhpA	1	1div	0.99	1div	1	0.70	1lis	2
1dosA	0.89	1dosA	1	0.99	1dosA	1	1ecmB	0.98	1ecmB	1	0.93	1ecmB	1
1ema	0.99	1ema	1	0.99	1ema	1	1exnB	0.99	1exnB	1	0.99	1exnB	1
1fiaB	0.98	1fiaB	1	0.99	1fiaB	1	1fkd	0.98	1fkd	1	0.89	2mhr	2
1flp	0.48	256bA	2	0.98	1flp	1	1fmb	0.41	1fmb	1	0.30	1tu1	2
1fmcA	1.00	1fmcA	1	1.00	1fmcA	1	1fna	0.99	1fna	1	0.95	1fna	1
1fps	0.99	1fps	1	1.00	1fps	1	1frb	0.99	1frb	1	0.99	1frb	1
1ft1A	1.00	1ft1A	1	1.00	1ft1A	1	1fua	1.00	1fua	1	0.99	1fua	1
1garB	0.97	1garB	1	0.96	1garB	1	1gen	0.99	1gen	1	0.98	1gen	1
1gky	0.97	1gky	1	0.55	1nfn	2	1gox	0.98	1gox	1	1.00	1ribA	9
1gpr	0.82	1gpr	1	0.87	1gtqA	50	1gtqA	0.99	1gtqA	1	0.99	1gtqA	1
1hfc	0.65	1hfc	1	0.84	153l	5	1htp	0.70	1jpc	2	0.72	1htp	1
1hus	0.99	1hus	1	0.94	256bA	4	1ido	0.99	1ido	1	0.99	1ido	1
1ifc	0.99	1ifc	1	0.99	1ifc	1	1iibA	0.24	1ycqA	27	0.50	1nsgB	15
1ipsA	0.99	1ipsA	1	0.99	1ipsA	1	1jdw	0.99	1jdw	1	0.99	1jdw	1
1jpc	0.95	1jpc	1	0.56	1xsoB	5	1knb	0.45	1amx	3	0.68	1ema	6
1kpf	0.47	1jpc	2	0.43	256bA	9	1ksaA	0.99	1ksaA	1	0.99	1cem	3
1lba	0.99	1lba	1	0.56	1lba	1	1lfb	0.22	1fiaB	4	0.22	1lfb	1
1lis	0.44	1pdo	2	0.99	1lis	1	1lkkA	0.98	1lkkA	1	0.98	1lkkA	1
1lrv	0.82	1rgp	2	0.92	1rgp	3	1lxa	0.42	2prk	5	0.97	1lxa	1
1mkaB	0.99	1mkaB	1	0.99	1mkaB	1	1msk	0.99	1msk	1	0.99	1msk	1
1mspB	0.37	1ravA	2	0.91	1ravA	10	1mugA	0.90	1mugA	1	0.92	1aep	5
1nar	0.99	1nar	1	0.99	1nar	1	1nbcB	0.64	1nbcB	1	0.86	1nbcB	1
1nfn	0.97	1nfn	1	0.99	1nfn	1	1nfp	0.99	1nfp	1	0.99	1nfp	1
1nls	0.99	1nls	1	0.99	1ema	4	1npk	0.27	1c52	26	0.34	1cex	26
1nsgB	0.59	2mhr	2	0.98	1nsgB	1	1nsj	0.78	1nsj	1	0.86	1nsj	1
1nsyA	0.99	1nsyA	1	1.00	1nsyA	1	1opy	0.37	2msbA	2	0.77	1opy	1
1oroA	0.99	1oroA	1	0.99	1oroA	1	1osa	0.99	1osa	1	0.79	1osa	1
1pdo	0.99	1pdo	1	0.94	1pdo	1	1phr	0.57	1bv1	3	0.97	1cgmE	6
1pmi	1.00	1pmi	1	0.99	1pmi	1	1pne	0.23	2rhe	25	0.42	1bkrA	12
1poh	0.99	1poh	1	0.99	1poh	1	1pud	1.00	1pud	1	1.00	1pud	1
1ravA	0.22	1fmb	3	0.85	1ris	22	1regY	0.53	1a62	2	0.78	2spcB	16
1rgeA	0.39	1acx	9	0.72	1lfb	17	1rgp	0.99	1rgp	1	0.92	1rgp	1
1rhs	1.00	1rhs	1	0.99	1rhs	1	1ribA	0.99	1ribA	1	1.00	1ribA	1
1rie	0.94	1rie	1	0.40	2mbA	16	1ris	0.81	1ris	1	0.99	1ris	1
1rkd	0.99	1rkd	1	0.73	1rkd	1	1rpa	1.00	1rpa	1	1.00	1rpa	1
1rsy	0.85	1rsy	1	0.42	1rsy	1	1sfp	0.85	1alx	4	0.97	1alx	5

TABLE I. (Continued.)

	Filtering			Viterbi				Filtering			Viterbi		
Sequence PDB code	Top family probability	Top family PDB code	Native family rank	Top family probability	Top family PDB code	Native family rank	Sequence PDB code	Top family probability	Top family PDB code	Native family rank	Top family probability	Top family PDB code	Native family rank
1smnB	0.82	1smnB	1	0.76	2cyp	2	1snc	0.39	1snc	1	0.99	2asr	12
1tig	0.99	1tig	1	0.45	1a32	2	1tlcB	0.99	1tlcB	1	0.99	1tlcB	1
1tm1	1.00	1tm1	1	1.00	1tm1	1	1tmy	0.98	1tmy	1	0.70	1tmy	1
1toh	0.99	1toh	1	0.60	1fps	2	1ttaB	0.58	1ttaB	1	0.99	1lis	5
1tul	0.40	1tu1	1	0.65	1ecmB	9	1uch	1.00	1uch	1	1.00	1uch	1
1udiI	0.82	1ycqA	4	0.70	1a32	12	1vhh	0.36	1pdo	9	0.24	1flp	8
1wab	0.99	1wab	1	0.99	1wab	1	1wgjB	0.99	1wgjB	1	0.96	1wgjB	1
1whi	0.61	1af5	2	0.38	1hus	11	1who	0.44	1fna	3	0.45	2end	3
1wpoB	0.99	1wpoB	1	0.71	1lxa	5	1xaa	0.99	1xaa	1	0.89	1cem	2
1xsoB	0.99	1xsoB	1	0.99	1xsoB	1	1ycqA	0.22	1aazA	3	0.97	1ecmB	10
1ycsA	0.96	1ycsA	1	0.42	1amx	17	1yer	0.99	1yer	1	0.99	1yer	1
1ygs	0.99	1ygs	1	0.82	1ygs	1	1ytw	1.00	1ytw	1	1.00	1ytw	1
256bA	0.99	256bA	1	0.99	256bA	1	2a0b	0.83	2a0b	1	0.99	2a0b	1
2aacA	0.99	2aacA	1	0.99	2aacA	1	2aak	0.99	2aak	1	0.99	2aak	1
2acy	0.48	2acy	1	0.66	1bkrA	5	2asr	0.99	2asr	1	0.99	2asr	1
2bopA	0.78	2bopA	1	0.68	1fiaB	2	2cba	0.97	2cba	1	0.83	1lrv	2
2cpl	0.99	2cp1	1	0.99	2cp1	1	2cyp	0.99	2cyp	1	1.00	2cyp	1
2dkb	0.99	2dkb	1	0.99	2dkb	1	2dri	0.48	1nsj	8	0.99	2dri	1
2end	0.97	2end	1	0.95	2end	1	2hts	0.33	2hts	1	0.26	2end	4
2lbd	1.00	2lbd	1	1.00	2lbd	1	2mhr	0.65	2hts	2	0.91	2mhr	1
2msbA	0.85	2msbA	1	0.66	1lfb	3	2phy	0.39	1opy	3	0.93	1opy	4
2plc	0.99	2plc	1	0.85	1rgp	3	2prk	0.99	2prk	1	0.64	1cem	2
2pth	0.55	1npk	2	0.93	2pth	1	2rhe	0.66	2rhe	1	0.23	1cmcA	10
2rn2	0.99	2rn2	1	0.99	2rn2	1	2sak	0.91	2sak	1	0.99	2sak	1
2sicI	0.91	2sicI	1	0.88	2sicI	1	2sil	0.99	2sil	1	0.91	2sil	1
2sniI	0.82	2sniI	1	0.96	2sniI	1	2spcB	0.99	2spcB	1	0.99	2spcB	1
2stv	0.97	2stv	1	0.48	2stv	1	3b5c	0.68	1fkd	6	0.64	1a32	22
3bct	0.99	3bct	1	0.99	3bct	1	3cla	0.99	3cla	1	0.37	1aep	4
3lip	0.65	1ako	9	0.99	1cem	5	4fgf	0.97	4fgf	1	0.75	1vhh	6
4xis	1.00	4xis	1	0.99	4xis	1	6fd1	0.77	6fd1	1	0.76	6fd1	1

[†]The structure prediction was done at the structural family level. For this structure-model library it is the same as SCOP structural superfamily prediction since only one family represents each superfamily. Top family indicates the most probable family according to the posterior probability value.

either fixed in length or will be assigned very low length variation probabilities. On the other hand, one can design a DSM to have similar length variations in all surface loops. The DSMs that we propose here are not trained but are built directly from the 3D protein structures deposited in the PDB[22] in such a way as to allow for possible variations. Hidden states of the DSM represent states of structural positions. These states encode the secondary structure and the level of solvent exposure of a structural position. Each hidden (structural) state in the model is characterized by the amino acid preferences for this state: a structural profile. The advantage of the DSM representation over the structural profile representation[20] is the simple encoding of the structural variations possible among structures with the same fold. These variations are usually the variable length of the secondary structure elements and alternative loop types (tight turn, turn, or coil) or loops with variable lengths connecting the secondary-structure elements.

Since the structure models that are derived from determined protein structures are not independent, a large fold-model library requires a method that systematically addresses the problem of hierarchical classification of protein structures (structure models). Thus when calculating the posterior probabilities, $P(\text{Model}\,|\,\text{seq})$, which involves the normalization over all models from the library, one needs to account for the similarities among models at each level of the hierarchy. Here we adopt the SCOP structural hierarchy. For example, for a fold represented by two superfamilies, each populated by four structural families, the prior probability for each family model would be $P(\text{Model}) = 1/2 \times 1/4$. The posterior probability of observing a particular structure model, given the sequence, is defined according to Bayes' rule: $P(\text{Model}\,|\,\text{seq}) = P(\text{Model}) \times P(\text{seq}\,|\,\text{Model})/P(\text{seq})$. In fold-recognition methods, the posterior normalization of the structure-model probabilities avoids overestimating the probability of observing a structural fold that is represented by many structures when compared to the probability of the fold that is represented by only one structure.

In a set of experiments we compared the performance of two fold-recognition methods. The first method identifies the best structure model for a sequence by using the probability of the optimal sequence-to-structure alignment. The second method identifies the best structure model for a sequence by using the total sequence-to-

structure alignment probability. Our results demonstrate that the total probability method predicts the structure model compatible with a sequence 40% more accurately than the optimal alignment probability method. For both methods we used the hierarchical posterior normalization of probabilities of structure models.

MATERIALS AND METHODS

DSM Structure Models

We constructed our DSM library by selecting 188 protein structures from the SCOP database.[23] These proteins were selected from SCOP (release pdb40d_1.38) representatives that have less than 40% sequence identity among themselves. We eliminated structures classified as irregular, engineered, or membrane-protein from the original set of proteins provided by SCOP. We additionally restricted proteins to one representative per SCOP structural superfamily and to single structural domains. The PDB identifiers for all the structures are given in Table I.

Each DSM is represented by three matrices: Φ, $\mathbf{H}$, and $\mathbf{x}_1$. The transition matrix Φ holds the conditional transition probabilities $\phi(s|s')$ of passing from each structural state s' to state s. The matrix $\mathbf{H}$ holds the conditional probabilities $h(a|s)$ of amino acid residue a being observed in (or emitted by) a structural state s. The initial state-distribution matrix $\mathbf{x}_1$ is a vector holding the probabilities $x_1(s)$ that the Markov chain starts in any state s at the beginning of the sequence.

Our structure models comprise positions that have the secondary structure (SS) assigned by DSSP[24] as helix or strand. One-residue kinks in helices are smoothed over and assigned helix secondary structure. The distance between the end positions of consecutive secondary structure elements is recorded and used to determine if tight turn or beta-turn loops are geometrically possible connections between consecutive elements. Structural positions are constructed from the backbone atoms and the beta carbon (C_β) or modeled C_β for the positions occupied by glycine in the native structure. Each structural-position environment is described by its secondary structure and Eisenberg-like solvent exposure of the position.[25] Solvent exposure is calculated for the poly-alanine chain and is independent of amino acids present in the native structure. By using the solvent exposure value, we define three solvent-exposure states: buried, partially buried, and exposed. Thus we have six types of structural states in all, counting positions in helix or strand. The possible loop states are tight turn (two-residue loops), beta turn (four residue loops), and coil loops longer than four residues.

An additional complication for the model building is posed by the fact that many protein structures are solved as a complex of a protein bound to a cofactor or other protein. Cofactors are often bulky and the solvent exposure pattern of a protein structure differs substantially, depending upon whether it is considered alone or with the cofactor bound. Thus for some of the 188 single-domain proteins, we built more than one structural DSM. If the protein structure is present in the PDB as a dimer then two models were built: one with the solvent exposure calculated for a dimeric structure and the second with the solvent exposure calculated for a monomeric structure. The number of constructed models exceeded two when there was more than one cofactor bound to the protein. Using this procedure we generated 350 DSMs from 188 protein structures.

These DSMs are built directly from protein structures deposited in the PDB by representing the structural positions as hidden states. An encoding of structural elements onto a DSM is shown in Figure 1. The secondary structure element, a β-strand or an α-helix, with n structural positions is represented as a strand or helix module. Each module of this top-level DSM is itself a DSM that starts and ends with a junction. The junctions do not emit amino acids. The observed variations among homologous structures are encoded by allowing an extension/deletion of the secondary structure by one position. An example of a strand module is shown in Figure 2. A helix module is constructed in a similar manner. The allowed loop-length variations are encoded in a generic loop module. A priori, three loop types are equally likely to connect any two consecutive SS elements: a tight turn, a beta turn, and a random coil. However, the analysis of observed loops indicates that if the distance between the end of the first SS element and the beginning of the following one is greater than 4.3 Å, no tight turn is possible. When the distance is greater than 10.5 Å, no beta turn is possible. Thus relative geometry of neighboring SS elements determines the possible loop types for a particular structure model. The loop module is shown in Figure 3.

The conditional probabilities of observing different amino acids given the structural state of the residue position are obtained from statistics on a large set of representative structures (unpublished data). These probabilities are independent of the particular structure being modeled. The transition probabilities from one structural state to another are selected to span or cover the expected variations in the homologous structures. Thus the probability of the SS element having the same length as the native structure is modeled by a transition probability of 1/3, the shortening of an SS element by one position has a probability of 1/3, and the extension of an SS element by any one of three solvent-exposure states has a probability of 1/9 (see Fig. 2). This a priori assignment of the transition probabilities differs from the usual HMM-building approach where the transition probabilities are trained by using a set of representative structures and proteins. Training of such HMMs is not feasible for folds having only one representative structure.

Posterior Probabilities of Models

The fold-recognition problem can be simply formulated as finding the posterior probabilities of different structure models given the query sequence: P(Model|seq). All available structure models define a model library. Two probabilities for observing a sequence, given a model, can be calculated using well-known algorithms.[19,26] First, the P(seq|Model,optimal-alignment) is the probability of observing a sequence, given a model and an optimal sequence-

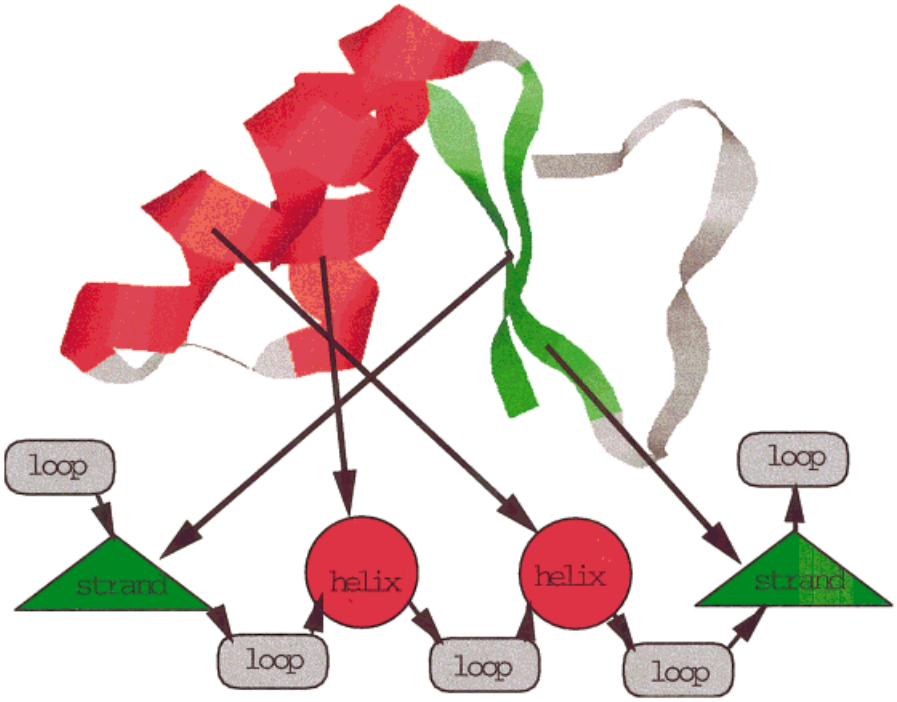

Fig. 1. Encoding of a structure from the PDB into DSM modules. Each structural element is represented by a building module: strand, helix, or loop. The internal parameters of each module, such as the number and type of hidden states and transition-matrix probabilities, are derived from the structural information as described in text.

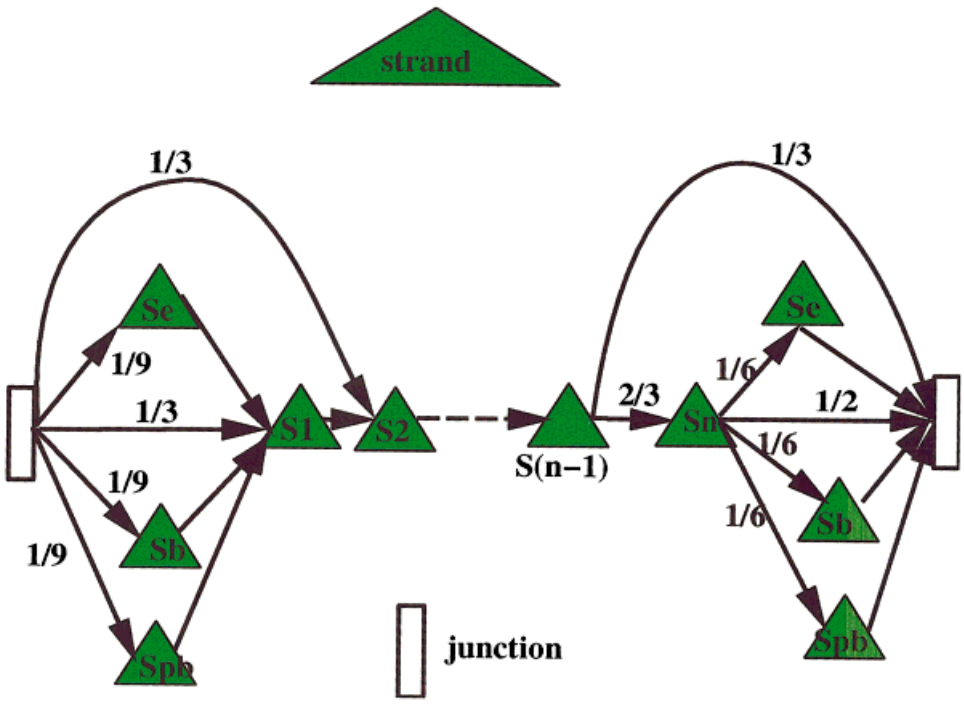

Fig. 2. A DSM strand module created from a strand of length n. Structural hidden states for strand residues are represented as triangles. “Si” denotes the solvent exposure state of the i-th strand position in the native structure. “Se” denotes an extension of the strand by an exposed strand position. “Sb” denotes a buried strand position and “Spb” denotes a partially buried strand position. Arrows connecting states represent the nonzero transition-matrix elements and numbers assigned to each line represent the transition probabilities. The arrows with no numbers associated with them have a transition probability equal to one.

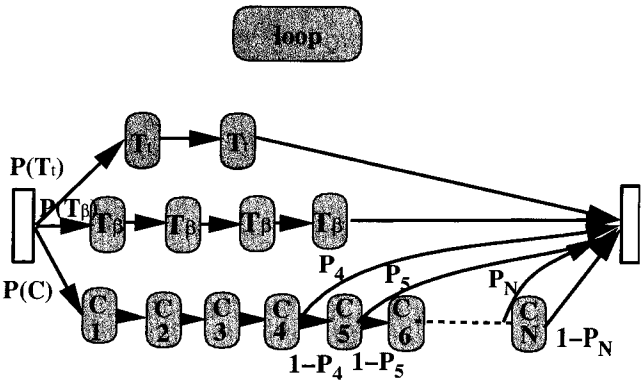

Fig. 3. A generic DSM loop module. Three types of loops connecting the secondary structure elements are possible: tight turn (T_t), beta turn (T_β), and coil or irregular loop (C). Arrows connecting states represent the nonzero transition-matrix elements and numbers assigned to each arrow represent the transition probabilities. The arrows with no numbers associated with them have a transition probability equal to one. The transition probabilities $P(T_t)$, the probability of a tight turn, $P(T_\beta)$, probability of a beta turn, or P(C), probability of coil are determined from the geometry of consecutive SS elements. If all three loops are possible $P(T_t) = P(T_\beta) = P(C) = 1/3$. If a tight turn is not allowed, $P(T_\beta) = P(C) = 1/2$ and $P(T_t) = 0$. The transition probabilities PN, 1-PN for coil loop states are set to represent a uniform loop-length distribution between minimal loop length L_{min} and maximal loop length L_{max}. For loop lengths greater than maximum ($L_{max} < N$) the loop-length distribution drops off exponentially. When only a coil is allowed, the minimal loop length is determined by the distance between the ends of consecutive SS elements; otherwise it is set to four when additionally a beta turn is allowed and it is set to two when a tight turn is also allowed. When the native-structure loop is shorter than ten residues L_{max} is set to ten. Otherwise L_{max} is equal to the length of the native-structure loop. The N- and C-terminal loop modules have the $P(T_t) = P(T_\beta) = 0$ and the loop can have zero length. The loop module does not contain any information about the solvent exposure of loop positions.

to-structure-model alignment (optimal path through the HMM). In HMM terminology, the algorithm that finds the optimal alignment (the most probable path) is called the Viterbi algorithm.[27] For all practical purposes, the Viterbi algorithm is equivalent to dynamic programming alignment algorithms.[28] The optimal alignment probability can be interpreted as the minimal "free energy" of the sequence in the conformation that is defined by the optimal alignment or path through the model. Second, the P(seq | Model) is the total probability of observing a sequence, given all possible alignments or paths through the model. In HMM terminology, the algorithm that calculates the probability summed over all sequence-to-structure-model alignment is called the Filtering algorithm.[19,26] Mathematically the P(seq | Model) gives a rigorous probability of observing a sequence, given the structure model.[13] We use the implementations of the Viterbi and the Filtering algorithms as described in references 13,19.

The posterior probabilities of observing a model given the sequence are calculated according to the Bayesian formula:

$$P(\text{Model}|\text{seq}) = \frac{P^*(\text{seq}|\text{Model}) \cdot P(\text{Model})}{P(\text{seq})} \quad (1)$$

$P(\text{seq}) = P(a_i)$, where $P(a_i)$ is the probability of observing the amino acid a_i in soluble proteins. $P^*(\text{seq}|\text{Model})$ indicates that we may use either the optimal-alignment probability or the total alignment probability. P(Model) is a prior probability of observing the Model. The probabilities are normalized according to the equation:

$$\sum_{\text{Models}\in\text{Library}} P(\text{Model}|\text{seq}) = 1 \quad (2)$$

The prior probabilities for each structure model in the library are assigned following a structural classification hierarchy. In our library we have models belonging to one of four structural classes: α, β, α/β, or α + β. Each class is assigned a prior probability of 0.25. Each class is represented by a number of SCOP structural folds = # folds/class. Each fold is represented by a number of SCOP structural superfamilies = # superfamilies/fold. Each superfamily is represented by a number of SCOP structural families = # families/superfamilies. Each family is represented by a number of Discrete Space Models that belong to the family = # models/family. Our library contains multiple models that were constructed from the same PDB entry with differences between DSMs that result from alternative solvent exposure estimates, as described above. Thus even if there is only one PDB entry that represents a SCOP superfamily, a structural superfamily may be represented by more than one DSM. The prior probability of a DSM classified by its class, fold, superfamily and family is

$$P(\text{Model}) = 0.25 \times \frac{1}{\#\text{folds/class}} \times \frac{1}{\#\text{superfamilies/folds}} \times \frac{1}{\#\text{families/superfamily}} \times \frac{1}{\#\text{models/family}} \quad (3)$$

By using the hierarchically-assigned model priors, we can rigorously answer the question: what is the probability of observing a unique structural superfamily given a sequence? The posterior probability of observing a unique structural superfamily given a sequence is

$$P(\text{superfamily}|\text{seq}) = \sum_{\text{Model}\in\text{family}\in\text{superfamily}} P(\text{Model}|\text{seq}) \quad (4)$$

Analogous equations apply for the posterior probabilities calculated for any level of the structural hierarchy: class, fold, superfamily, or family.

We use a simple binary decision rule: either one model is preferred in comparison to all others or not. Thus we accept only the top folds/models with the posterior probability greater than 0.5.

RESULTS AND DISCUSSION

We compared two alternative methods of fold recognition. In the first, the Viterbi method, we used the value of the optimal sequence-to-model alignment probability as calculated by the Viterbi algorithm. In the second, the Filtering method, we used the value of the total probability as calculated by the Filtering algorithm. For both methods, we used the hierarchical prior model probabilities to calculate the normalized posterior probability values and to make a fold prediction.

In the first set of experiments, we calculated P(family | seq), the posterior probability for each structural family, for 188 native sequences on 350 DSMs from our library. The fold prediction results for the Viterbi and for the Filtering method are presented in Table I. The posterior probabilities of observing a structural family given the sequence were normalized according to Eqs. 1, 2, and 3. With the Filtering fold-recognition method, 152 out of 188 sequences ranked the native structural family with the highest probability. With the Viterbi fold-recognition method, 110 out of 188 sequences ranked the native structural family with the highest probability. The "acceptable" predictions with the top structural family that had a probability of at least 0.5 had the following results. For the Filtering fold-recognition method, there were 161 predictions and 145 were correct: a 90% success rate. For the Viterbi fold-recognition method, there were 188 predictions and 107 were correct: a 64% success rate. These results demonstrate that the Filtering fold-recognition method is 40% more accurate than the Viterbi fold-recognition method.

In the second set of experiments, we tested performance of the fold-recognition methods in recognizing the structures of proteins that do not necessarily share sequence similarity with the proteins used to generate our set of 350 DSMs but nevertheless have similar structural folds. For testing, we used a set of proteins classified into ten SCOP structural folds that were recently used for testing the Recursive Dynamic Programming (RDP) threading fold-recognition method.[11] We removed the cysteine-knot cytokines fold because it is an irregular fold with very few secondary-structure elements. Our DSMs are based primarily on secondary-structure and solvent-exposure preferences and such irregular structures do not produce specific DSMs. In these experiments, we calculated the posterior probability P(fold | seq) for each structural fold for 71 sequences listed Table II on our library of 350 DSMs. The fold-recognition results for the Viterbi and for the Filtering method are presented in Table II. With the Filtering fold-recognition method, 33 out of 71 sequences ranked the correct structural fold with the highest probability. With the Viterbi method, only 31 out of 71 sequences ranked the correct structural fold with the highest probability.

When prediction is restricted to a posterior probability of at least 0.5, the Filtering method predicted 32 of 53 correctly: a 60% success rate. The Viterbi method predicted only 29 of 66 correctly: a 44% success rate. These results confirm that the Filtering fold-recognition method is 40% more accurate than the Viterbi fold recognition. The Filtering fold-recognition rate of 60% is slightly better than the 57% reported for RDP threading.[11]

The worst fold-recognition rate by the Filtering method was obtained for the α/β hydrolases (a 0% fold-recognition rate), represented in our library by only one model structure (3lip) and for the viral coat and capsid protein fold (17% correct) represented in our library by only one model structure (2stv). Both of these folds are adopted by proteins with highly variable sequence lengths as reported by Thiele et al.[11] The length of α/β hydrolases varies from 265 to 534 amino acids and the length of viral coat proteins varies from 175 to 548 amino acids. Our DSM variations in the secondary-structure segment and loop lengths are not large enough to accommodate such differences. Thus it is not surprising that having only one representative structure for such diverse folds limits the fold-recognition method severely. These results call for the expansion of our model library to include more nonhomologous representatives for each structural family.

The most frequently used test set for the fold-recognition methods is the UCLA1 benchmark proposed by Fischer and Eisenberg.[29] We compared our fold-recognition method with the results of the GenTHREADER reported recently by Jones in reference 12. The GenTHREADER was one of three top-ranked methods during the recent CASP3 fold-recognition contest.[1] It thus appears to be a very good representative of other fold-recognition methods. The UCLA1 benchmark comprises a library of 296 representative structures and a set of 68 pairs of proteins with low sequence similarity, which are supposed to be identified by fold recognition. Of these 68 pairs, only 44 have a target structure classified by both SCOP and CATH as a single structural domain. It should be noted that many of the benchmark pairs represent functionally related proteins, since with significant sequence similarity, they share at least one functional domain.

Like most fold-recognition methods, GenTHREADER includes a sequence similarity measure between the template structure and the query sequence. Such methods use a dynamic programming algorithm to align a query sequence to a structural template and do not require that every amino acid in the sequence be represented by a state from a structural template. Thus a very short sequence, representing a single domain protein can be aligned to a model of a very long multidomain protein and vice versa. In contrast, the DSMs that we propose here require that every amino acid from the query sequence is emitted by (or aligned to) some modeled structural state. The structure of the DSMs permits threading of a sequence onto a model of comparable length. In particular, sequences or partial sequences that are shorter than the minimum length path through the model have a prior probability set to zero. To be consistent with this framework, we examined only those protein pairs from the UCLA1 benchmark that represent single-domain structures. This leaves for consideration 44 pairs with PDB structure codes: 1bovA, 1cauA, 1ede, 1ego, 1hbg, 1ifc, 1lfb, 1molA, 1nsbA, 1paz, 1rbp, 1rnh, 1shaA, 1tca, 1ubq, 1ycc, 256bA, 2ayh, 2ccyA, 2cpp, 2fox, 2fxb, 2gmfA, 2hipA, 2plv1, 2rhe, 2scpA, 2tbvA, 2trxA, 3hlaB, 4cla, 7rsa, and 9rnt (few target structures occur more than once in the benchmark). From those 44 query sequence-target structure pairs, 6 (3cd4-2rhe, 1dsbA-2trxA, 1cid-2rhe, 1crl-1ede, 1bgeB-2gmfA, and 1gp1A-2trxA) had query sequences too long to fit onto the assigned target model. Three of those query sequences are two-domain proteins and the last three represent the SCOP superfamily member with a sequence too long for the target protein model. We were forced to limit the set of 68 protein pairs to the 38 listed Table III. For those pairs,

TABLE II. Results of the Fold-Recognition Experiments for Ten SCOP Structural Folds[†]

		Fold-recognition method					
		Filtering			Viterbi		
Sequence PDB code	Correct fold PDB code	Top fold probability	Top fold PDB codes	Correct fold rank	Top fold probability	Top fold PDB codes	Correct fold rank
SCOP fold classification: OB fold							
1gpc	1a62	0.34	1aerB	14	0.97	1lxa	39
1snc	1a62	0.39	1gpr 1htp	2	0.94	4-helix bundle	10
1prtF	1a62	0.42	1lfb 2hts	40	0.78	4-helix bundle	32
1prtD	1a62	0.47	2sicI	38	0.39	2end	13
1a62	1a62	0.77	1a62 1snc 1wgjB	1	0.69	1cis	5
1pyp	1a62	0.99	1a62 1snc 1wgjB	1	0.84	1a62 1snc 1wgjB	1
1wgjA	1a62	0.99	1a62 1snc 1wgjB	1	0.94	1a62 1snc 1wgjB	1
2prd	1a62	0.99	1lxa	20	0.63	α/α superhelix	34
SCOP fold classification: four-helical cytokines							
1lki	1bgc	0.71	1bgc	1	1.00	1bgc	1
1ilk	1bgc	0.96	4-helix bundle	23	1.00	4-helix bundle	4
1huw	1bgc	0.97	1ahq	21	0.77	1flp	3
1bgc	1bgc	0.99	1bgc	1	1.00	1bgc	1
SCOP fold classification: globin-like							
1eca	1flp	0.35	1ahq	36	0.54	1flp	1
1cpcA	1flp	0.39	1hfc	41	0.70	4-helix bundle	2
1pbxA	1flp	0.45	1cewI 1opy 1udiI	10	0.94	1flp	1
2gdm	1flp	0.50	1flp	1	0.85	4-helix bundle	2
2fal	1flp	0.66	1flp	1	0.91	1flp	1
2hbg	1flp	0.67	1ae9B	23	0.45	1flp	1
1flp	1flp	0.75	1flp	1	0.99	1flp	1
1hlb	1flp	0.85	1flp	1	0.51	1flp	1
1hrm	1flp	0.96	1flp	1	0.99	1flp	1
1ash	1flp	0.97	1flp	1	0.99	1flp	1
3sdhA	1flp	0.99	1a62 1snc 1wgjB	57	0.92	1flp	1
1cpcB	1flp	0.99	1flp	1	1.00	1flp	1
SCOP fold classification: lipocalins							
1mup	1ifc	0.30	1ifc	1	0.61	1aep	31
1bbpA	1ifc	0.38	1knb	2	0.22	1cex 1tmy 1wab	17
1epaA	1ifc	0.92	1ifc	1	0.80	1aep	17
1hbq	1ifc	0.96	1ble	3	0.85	1aep	39
1ifc	1ifc	0.99	1ifc	1	1.00	1ifc	1
1hmt	1ifc	0.99	1ifc	1	1.00	1ifc	1
SCOP fold classification: α/β TIM-barrel							
1ubsA	1nar	0.24	1rhs	8	0.95	2cyp	19
1fbaA	1nar	0.41	2dri	2	1.00	1fps	7
5tima	1nar	0.49	1chd	3	0.68	1cem	2
1pbgA	1nar	0.49	1jdw	4	0.65	1ribA	5
1xyzA	1nar	0.53	α/β TIM-barrel	1	1.00	1ribA	9
1oyc	1nar	0.64	1rpa	2	0.62	1fps	7
1byb	1nar	0.92	1alkA	6	1.00	α/α superhelix	5
1nar	1nar	0.99	α/β TIM-barrel	1	1.00	α/β TIM-barrel	1
1nfp	1nar	0.99	α/β TIM-barrel	1	1.00	α/β TIM-barrel	1
2ebn	1nar	0.99	α/β TIM-barrel	1	0.99	α/β TIM-barrel	1
2acq	1nar	1.00	α/β TIM-barrel	1	1.00	α/β TIM-barrel	1
SCOP fold classification: four-helical up-and-down bundle							
2hmzA	1nfn	0.53	4-helix bundle	1	1.00	4-helix bundle	1
2tmvP	1nfn	0.76	1gpr 1htp	9	0.58	4-helix bundle	1
1was	1nfn	0.99	4-helix bundle	1	1.00	4-helix bundle	1
2ccyA	1nfn	0.99	4-helix bundle	1	0.87	2spcB	2
11pe	1nfn	0.99	4-helix bundle	1	1.00	4-helix bundle	1
1nfn	1nfn	0.99	4-helix bundle	1	1.00	4-helix bundle	1
256bA	1nfn	0.99	4-helix bundle	1	1.00	4-helix bundle	1
SCOP fold classification: flavodoxin-like							
3tmy	1tmy	0.94	1cex 1tmy 1wab	1	0.68	1cex 1tmy 1wab	1

TABLE II. (Continued.)

		Fold-recognition method					
		Filtering			Viterbi		
Sequence PDB code	Correct fold PDB code	Top fold probability	Top fold PDB codes	Correct fold rank	Top fold probability	Top fold PDB codes	Correct fold rank
3chy	1tmy	0.99	1cex 1tmy 1wab	1	0.92	1cex 1tmy 1wab	1
2fox	1tmy	0.99	1cex 1tmy 1wab	1	0.74	1cex 1tmy 1wab	1
1cus	1tmy	0.99	1cex 1tmy 1wab	1	1.00	1cex 1tmy 1wab	1
1rcf	1tmy	0.99	1mugA	4	0.80	1bgc	12
SCOP fold classification: viral coat and capsid proteins							
4rhv3	2stv	0.45	3cla	4	0.88	1ema	5
1bbt3	2stv	0.48	1cghA	29	0.75	1531	21
2bpa2	2stv	0.54	1ba7A 4fgf	21	0.78	1ema	17
1bbt1	2stv	0.63	1gky	38	0.96	4-helix bundle	42
1bbt2	2stv	0.84	1am2	5	1.00	1ema	5
2stv	2stv	0.97	2stv	1	0.54	2stv	1
SCOP fold classification: α/β hydrolases							
3tg1	3lip	0.28	1wpoB	48	0.68	ferredoxin-like	49
1ede	3lip	0.42	1tm1	22	1.00	21bd	24
1thtA	3lip	0.62	a/b TIM-barrel	8	0.45	1cem	16
1tahB	3lip	0.63	1av6A	5	0.85	α/α superhelix	4
3lip	3lip	0.66	1ako	6	1.00	1cem	5
1tca	3lip	0.82	2prk	18	0.91	1cby	18
SCOP fold classification: ferrodoxin-like							
1regX	6fd1	0.29	1a62 1snc 1wgjB	7	0.93	2spcB	9
1aps	6fd1	0.50	β sandwich	3	0.69	1fiaB	4
2bopA	6fd1	0.61	Ferrodoxin-like	1	0.85	1fiaB	2
6fd1	6fd1	0.62	1kpf	2	0.45	Ferredoxin-like	1
1nhkR	6fd1	0.74	1pdo	8	0.94	1aep	32
1pba	6fd1	0.79	1a68	13	0.72	1a32	3

[†]The structure prediction was done at the structural fold level. Top fold indicates the most probable structural fold according to the posterior probability value. For folds represented by more than three PDB entries, we used the SCOP names as follows: α/β TIM-barrel: 1add, 1aj2, 1amk, 1dhpA, 1dosA, 1frb, 1gox, 1nar, 1nfp, 1nsj, 1pud, 2plc and 4xis. 4-helix bundle. 1cgmE, 1nfn, 1nsgB, 256bA, 2a0b, 2asr and 2mhr. Ferredoxin-like: 1ab8B, 1npk, 1regY, 1ris, 2acy, 2bopA and 6fd1. β sandwich (Immunoglobuline-like): 1acx, 1fna, 1mspB, 1xsoB and 2rhe. α/α superhelix: 1a17, 1ft1A, 1lrv and 3bct.

the query sequence can be threaded onto the "correct" DSM that represents a single structural domain.

By using the software described in Methods, we have built a DSM library from the benchmark library of structures. Since most of the benchmark pairs represent proteins from the same functional family, the sequence similarity between the query and the target could play a crucial role in fold recognition. Thus for each target protein, we have created additional DSMs with an embedded minimal pattern of residues conserved within the functional family. Pattern embedding in a structural DSM was described previously.[30] In each model, a structural state of the position occupied by the strictly-conserved residue was replaced by a state that represented the conserved amino acid. The probability of emitting any other amino acid from the conserved position was set to zero. Only the structural positions that were located in a strand or a helix or at the start/end position of a strand or helix had the conserved-residue state embedded. Here, we did not design a separate model for the conserved residues embedded in a loop DSM module. Thus, functionally conserved residues in loop positions were not included.

Different target protein families have different numbers of conserved positions. Thus for each structure, we built a number of minimal-pattern-embedded DSMs. Such a procedure creates many models for some PDB structures and very few for others. Additionally, the UCLA1 benchmark library consists of proteins that are either single- or multi-domain proteins. Thus the task of assigning the model priors for the UCLA1 benchmark is not as straightforward as for our standard DSM library that represents only single structural domains. To avoid bias between the underrepresented and overrepresented structural domains and folds, each sequence-to-PDB-structure score was selected as the best score among DSMs made from that PDB structure. This method of scoring is equivalent to the scoring method used by GenTHREADER and other fold-recognition methods where no prior information about the fold or superfamily representation is included.

In Table III we present detailed results of the Filtering fold-recognition experiment done for the 38 protein pairs threaded through the library of structural and minimal-pattern-embedded DSMs created from the benchmark library of structures. For each query sequence, we report

TABLE III. Results of the Filtering Fold-Recognition Method for the UCLA1 Benchmark and the Library of the Minimal Pattern Embedded DSMs[†]

Query sequence		Target structure					Top ranked structure		
PDB	Length	PDB	Length	Prob	Rank	Patt	PDB	Length	Prob
1aaj	105	1paz	123	0.117	3	3	1fkf	107	0.330
1aba	87	1ego	85	0.972	1	2	1ego	85	0.972
1aep[b]	161	256ba	106	0.000	2	1	2cy3	118	0.999
1bbha	131	2ccya	128	0.999	1	3	2ccyA	128	0.999
1bbt1	213	2plv1	302	0.987	1	5	2plv1	302	0.987
1c2ra	116	1ycc	108	0.299	2	1	1ego	85	0.311
1caub	184	1caua	181	0.586	1	3	1cauA	181	0.586
1cewi[a]	108	1mola	94	0.578	1	0	1molA	94	0.578
1dxtb	147	1hbg	147	0.692	1	2	1hbg	147	0.692
1eaf	243	4cla	213	0.999	1	2	4cla	213	0.999
1fxia	96	1ubq	76	0.340	1	2	1ubq	76	0.340
1hip	85	2hipa	72	0.036	7	3	1ubq	76	0.270
1hom	68	1lfb	99	0.963	1	2	11fb	99	0.963
1hrha	136	1rnh	155	0.766	1	9	1rnh	155	0.766
1isua	62	2hipa	72	0.739	1	3	2hipA	72	0.739
1ltsd	103	1bova	69	0.795	1	0	1bovA	69	0.795
1mdc	132	1ifc	132	0.999	1	0	1ifc	132	0.999
1mup	166	1rbp	182	0.984	1	4	1rbp	182	0.984
1onc	104	7rsa	124	0.934	1	3	7rsa	124	0.934
1pfc	113	3hlab	99	0.973	1	2	3hlaB	99	0.973
1rcb	129	2gmfa	127	0.285	2	2	2end	138	0.634
1saca[a]	204	2ayh	214	0.054	2	0	5ptp	223	0.877
1stfi[a]	98	1mola	94	0.442	1	0	1molA	94	0.442
1taha[a]	318	1tca	317	0.713	1	7	1tca	317	0.713
1tie[a]	172	4fgf	146	0.947	1	6	4fgf	146	0.947
1tlk[a]	154	2rhe	114	0.926	1	3	2rhe	114	0.926
2azaa	129	1paz	123	0.253	1	3	1paz	123	0.253
2hpda	471	2cpp	414	1.000	1	1	2cpp	414	1.000
2mtac	147	1ycc	108	0.000	24	1	1lz1	130	0.801
2pna	104	1shaa	104	0.963	1	2	1shaA	104	0.963
2sara	96	9rnt	104	0.232	1	3	9rnt	104	0.232
2sas	185	2scpa	174	0.999	1	2	2scpA	174	0.999
2sim	381	1nsba	390	0.712	1	6	1nsbA	390	0.712
3chy[a]	128	2fox	138	0.000	16	1	2gmfA	127	0.426
3hlab[a]	99	2rhe	114	0.860	1	3	2rhe	114	0.860
4sbva	261	2tbva	387	0.009	13	5	1tfd	304	0.324
5fd1	106	2fxb	81	0.717	1	3	2fxb	81	0.717
8i1b	152	4fgf	146	0.334	2	6	1ptsA	121	0.376

[†]Columns report: the benchmark query sequence PDB code and its length (as listed the SEQRES records in the corresponding PDB file). For the target structure assigned by the benchmark the columns report: the PDB code, the sequence length, the probability of the corresponding DSM, the rank of the target structure and the number of conserved residues in the minimal pattern embedded in the DSM (column label "patt"). For the top-ranked structure columns report: the PDB code, the sequence length, and the probability of the corresponding DSM.

[a]The query sequences for which the benchmark target structure is the member of the same SCOP fold or SCOP superfamily but is the member of different functional family.

[b]The benchmark pair that has been assigned different SCOP fold but the same CATH fold.

the rank of the benchmark target structure. We report in Table IV the highest rank of the structure with the CATH fold assignment (first three numbers) identical to that of the query sequence, the CATH rank. Out of 38 query sequences, 28 recognized the correct target structure at rank 1 as defined by the benchmark and by CATH. The GenTHREADER procedure for this set (see Table IV) reported the following results: 26 out of 38 sequences have a benchmark target at rank 1 and 28 sequences had a CATH rank 1. Out of ten benchmark pairs that are classified by SCOP as members of a different functional family, the Filtering fold-recognition method identified seven with CATH rank 1, while GenTHREADER identified five with CATH rank 1. The Filtering fold-recognition method with the library of the minimal-pattern-embedded DSMs made 28 predictions (the probability of the top structure was greater than 0.5) and 24 of those predictions were correct.

Table IV compares the benchmark target rank obtained by different fold-recognition methods. In addition to the results reported for GenTHREADER in reference 12, we present the results for the Viterbi fold-recognition method. The Viterbi fold-recognition method was half as successful as the Filtering fold-recognition method. For the same

TABLE IV. Comparison of UCLA1 Benchmark Fold-Recognition Results[†]

		DSMs											
		with embedded minimal pattern						Structural only			GenTHREADER with sequence similarity		
PDB		Filtering			Viterbi			Filtering					
Query	Target	Top prob	Rank	CATH rank	Top prob	Rank	CATH rank	Top prob	Rank	CATH rank	Net	Rank	CATH rank
1aaj	1paz	0.330	3	3	0.525	6	6	0.426	70	4	1.000	1	1
1aba	1ego	0.972	1	1	0.585	2	2	0.969	40	40	1.000	1	1
1aep	256ba	0.999	2	2	0.936	4	4	0.999	3	3	0.802	4	4
1bbha	2ccya	0.999	1	1	1.000	1	1	0.780	5	4	1.000	1	1
1bbt1	2plv1	0.987	1	1	0.761	1	1	0.902	39	15	1.000	1	1
1c2ra	1ycc	0.311	2	2	0.487	3	3	0.436	10	10	1.000	1	1
1caub	1caua	0.586	1	1	0.579	33	33	0.372	14	14	1.000	1	1
1cewi	1mola	0.578	1	1	0.494	1	1	0.588	1	1	0.023	>100	>100
1dxtb	1hbg	0.692	1	1	0.999	1	1	0.225	17	17	1.000	1	1
1eaf	4cla	0.999	1	1	0.493	8	8	0.999	1	1	0.787	1	1
1fxia	1ubq	0.340	1	1	0.536	15	15	0.385	14	14	0.958	1	1
1hip	2hipa	0.270	7	3	0.955	9	9	0.199	24	24	1.000	1	1
1hom	1lfb	0.963	1	1	0.939	1	1	0.474	2	2	0.109	1	1
1hrha	1rnh	0.766	1	1	0.999	1	1	0.620	117	117	1.000	1	1
1isua	2hipa	0.739	1	1	0.971	1	1	0.477	12	12	0.928	1	1
1ltsd	1bova	0.795	1	1	0.542	24	24	0.826	1	1	0.130	6	6
1mdc	1ifc	0.999	1	1	0.999	1	1	0.999	1	1	1.000	1	1
1mup	1rbp	0.984	1	1	0.999	1	1	0.430	12	12	1.000	1	1
1onc	7rsa	0.934	1	1	0.996	1	1	0.271	29	29	1.000	1	1
1pfc	3hlab	0.973	1	1	0.981	2	2	0.791	5	1	1.000	1	1
1rcb	2gmfa	0.634	2	2	0.566	3	3	0.783	2	2	0.084	5	5
1saca	2ayh	0.877	2	2	0.608	36	2	0.948	2	2	0.224	98	5
1stfi	1mola	0.442	1	1	0.768	3	3	0.442	1	1	0.065	>100	>100
1taha	1tca	0.713	1	1	0.707	3	3	0.833	23	4	1.000	3	1
1tie	4fgf	0.947	1	1	0.650	6	6	0.383	6	6	0.019	>100	>100
1tlk	2rhe	0.926	1	1	0.804	19	19	0.327	18	13	0.217	1	1
2azaa	1paz	0.253	1	1	0.975	12	4	0.329	29	1	0.677	1	1
2hpda	2cpp	1.000	1	1	1.000	1	1	1.000	1	1	1.000	1	1
2mtac	1ycc	0.801	24	24	0.947	6	6	0.801	33	33	0.943	2	1
2pna	1shaa	0.963	1	1	0.802	7	7	0.312	3	3	1.000	1	1
2sara	9rnt	0.232	1	1	0.968	6	6	0.256	42	31	0.015	5	5
2sas	2scpa	0.999	1	1	0.869	1	1	0.999	1	1	1.000	1	1
2sim	1nsba	0.712	1	1	0.999	2	2	0.884	8	8	0.019	>100	>100
3chy	2fox	0.426	16	16	0.570	24	24	0.426	20	20	0.742	14	14
3hlab	2rhe	0.860	1	1	0.990	2	2	0.405	30	22	0.994	1	1
4sbva	2tbva	0.324	13	2	0.999	1	1	0.332	18	2	0.218	1	1
5fd1	2fxb	0.717	1	1	0.480	1	1	0.548	2	2	1.000	1	1
8ilb	4fgf	0.376	2	2	0.997	45	45	0.565	24	24	0.634	1	1

[†]For each query sequence, target structure we report in the column "rank," the rank of the benchmark assigned target. In the column "CATH rank," we report the rank of the highest ranking structure with the same CATH fold assignment as the query sequence. In the column "top prob," we report the probability of the top ranking structure. The column "Net" lists the network output confidence as reported for the GenTHREADER. Columns labeled "Filtering" correspond to the results of the Filtering fold-recognition method. Columns labeled "Viterbi" correspond to the results of the Viterbi fold-recognition method. For the GenTHREADER the rank of the structures ranking higher than 100 was reported only as >100.

library of minimal-pattern-embedded DSMs, the Viterbi method recognized at first rank only 14 benchmark targets. Conserved pattern embedding in the DSM greatly improved recognition of the UCLA1 benchmark targets, which should not be surprising since the benchmark was constructed to test a fold-recognition method that relied on sequence similarity.[29] Out of 38 benchmark pairs, 28 are members of the same functional family. When the library is restricted to structural DSMs, the Filtering fold-recognition method gives a prediction for 19 sequences (top probability greater than 0.5) and only 6 are correct targets according to the benchmark and 7 according to CATH fold assignment. We also note that the minimal pattern embedding helped in recognizing the correct benchmark target even when the original structural DSM contains very few secondary structure elements. For example, the 2hipA structure is classified as small irregular fold by SCOP, and the automatically constructed structural DSM contains only two beta strands. We exclude such models from our standard library, since they mostly comprise loop states that are amino acid non-specific. Nevertheless, including a pattern of only three conserved residues allowed one of the

query sequences (1isuA) to recognize the correct benchmark target with the highest probability.

The results of this comparison demonstrate that the calculation of proper P(Model|seq) DSM posterior probabilities by Filtering competes with other top ranking fold-recognition methods when a minimal sequence pattern of conserved residues is embedded in the DSM. Here, we embedded only a minimal pattern of strictly conserved residues that belong to secondary-structure positions assigned as strand or helix. From the success of the fold recognition that includes an amino acid similarity measure over the whole sequence-to-structure alignment,[1,12] it is apparent that combining the full-length positional amino acid profile of the structure with the DSM representation should further improve the performance of our fold-recognition method. Work on embedding the full-length positional profiles in DSMs is in progress.

CONCLUSION

In our fold-recognition method, we have incorporated the hierarchical structure classification scheme that allows a rigorous assignment of posterior fold/model probabilities. Each unique structural class represented by different DSMs has assigned a posterior probability for that particular class. The Bayesian assignment of the posterior probabilities systematically addresses the problem of interpreting fold-recognition results that use a library of diverse and interdependent models.

We have presented here a class of relatively simple structural Hidden Markov Models, i.e., the Discrete State Models. These models are built automatically from the protein structures deposited in the PDB. The DSMs represent amino acid preferences for a small set of structural states. Our DSMs encode only six SS/solvent-exposure structural states and three loop states. As such these models can be seen as an alternative and very simple representation of a structural profile. The HMM representation has two advantages over the structural profile representation used previously by many fold-recognition methods.[2,11] The first advantage of the HMM representation is the incorporation in those models of structural variations such as variable secondary-structure element length and variable loop states that connect the secondary-structure elements. The second advantage comes directly from HMM theory. The compatibility of the query sequence with a model can be rigorously calculated as the total (summed over all sequence-to-structure alignments) probability of the model. We have demonstrated that the fold-recognition method that uses the total probability is 40% more accurate than the "standard" fold-recognition method that uses the probability of the optimal sequence-to-structure alignment.

ACKNOWLEDGMENTS

We thank Jim White for many discussions about the HMMs, and continuing support for the DSMs project. We also thank Scott Mohr for carefully reading the manuscript and making many helpful suggestions.

REFERENCES

1. Murzin AG. Structure classification-based assessment of CASP3 predictions for the fold-recognition targets. Proteins 1999;Suppl 3:88–103.
2. Rost B, Schneider R, Sander C. Protein fold recognition by prediction-based threading. J Mol Biol 1997;270:471–480.
3. Bryant SH, Lawrence CE. An empirical energy function for threading protein sequence through the folding motif. Proteins 1993;16:92–112.
4. Miyazawa S, Jernigan RL. Estimation of effective interresidue contact energies from protein crystal structures: quasi-chemical approximation. Macromolecules 1985;18:534–552.
5. Godzik A, Skolnick J, Kolinski A. A topology fingerprint approach to the inverse folding problem. J Mol Biol 1992;227:227–238.
6. Sippl MJ. Knowledge-based potentials for proteins. Curr Opinion Struct Biol 1995;5:229–235.
7. Jernigan RL, Bahar I. Structure-derived potentials and protein simulations. Curr Opinion Struct Biol 1996;6:195–209.
8. Miyazawa S, Jernigan RL. Residue-residue potentials with a favorable contact pair term and unfavorable high packing density term, for simulation and threading. J Mol Biol 1996;256:623–644.
9. Taylor WR. Multiple sequence threading: an analysis of alignment quality and stability. J Mol Biol 1997;269:902–943.
10. Skolnick J, Jaroszewski L, Kolinski A, Godzik A. Derivation and testing of pair potentials for protein folding: when is the quasichemical approximation correct? Protein Sci 1997;6:676–688.
11. Thiele R, Zimmer R, Lengauer T. Protein threading by recursive dynamic programming. J Mol Biol 1999;290:757–779.
12. Jones DT. GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. J Mol Biol 1999;287: 797–815.
13. White JV, Stultz CM, Smith TF. Protein classification by stochastic modeling and optimal filtering of amino acid sequences. Bull Math Biosci 1994;119:35–75.
14. Lathrop RH et al. Analysis and algorithms for protein sequence-structure alignment. Salzberg S, Searls D, Kasif S, editors. Amsterdam, Netherlands: Elsevier Press; 1998. p 227–283.
15. Lathrop RH, Rogers Jr. RG, Smith TF, White JV. A Bayes-optimal sequence-structure theory that unifies protein sequence-structure recognition and alignment. Bull Math Biol 1998;60:1–33.
16. Levitt M. Competitive assessment of protein fold recognition and alignment accuracy. Proteins 1997;Suppl 1:92–104.
17. Reference deleted in proofs.
18. Russell RB, Copley RR, Barton GJ. Protein fold recognition by mapping predicted secondary structures. J Mol Biol 1996;259:349–365.
19. White JV. Bayesian analysis of time series and dynamic models. New York: Marcel Dekker; 1988. p 255–283.
20. Bowie JU, Luthy R, Eisenberg D. A method to identify protein sequences that fold into a known three-dimensional structure. Science 1991;253:164–170.
21. Park J et al.. Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. J Mol Biol 1998;284:1201–1210.
22. Bernstein FC et al. The protein data bank: a computer-based archival file for macromolecular structures. J Mol Biol 1977;112: 535–542; Brookhaven Protein Data Bank release 80.
23. Murzin A, Brenner SE, Hubbard T, Chothia C. SCOP: a structural classification of proteins database for the investigation of the sequences and structures. J Mol Biol 1995;247:536–540.
24. Kabsch W, Sander C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers 1983;22:2577–2637.
25. Lee B, Richards FM. The interpretation of protein structures: estimation of static accessibility. J Mol Biol 1971;55:379–400.
26. Rabiner LR. A tutorial on Hidden Markov Models and selected applications in speech recognition. Proc IEEE 1989;77:257–286.
27. Viterbi AJ. Error bounds for convolutional codes and an asymptotically optimal decoding algorithm. IEEE Trans Information Theory IT-1967;13:260–269.
28. Smith TF, Waterman MS. Identification of common molecular subsequences. J Mol Biol 1980;147:195–197.
29. Fischer D, Eisenberg D. Protein fold recognition using sequence-derived predictions. Protein Sci 1996;5:757–779.
30. Yu L, White JV, Smith TF. A homology identification method that combines sequence and structure information. Protein Sci 1998;7: 2499–2510.