Published in final edited form as:
J Mol Biol. 2009 March 27; 387(2): 431–450. doi:10.1016/j.jmb.2008.12.044.

Structural Alphabets for Protein Structure Classification: a Comparison Study

Quan Le[a], **Gianluca Pollastri**[a], and **Patrice Koehl**[b,*]

[a]Complex and Adaptive Systems Laboratory School of Computer Science and Informatics University College Dublin Dublin, Ireland [b]Department of Computer Science and Genome Center University of California, Davis Davis, CA 95616, USA

Abstract

Finding structural similarities between proteins often helps revealing shared functionality which otherwise might not be detected by native sequence information alone. Such similarity is usually detected and quantified by protein structure alignment. Determining the optimal alignment between two protein structures remains however a hard problem. An alternative approach is to approximate each protein 3D structure using a sequence of motifs derived from a structural alphabet. Using this approach, structure comparison is performed by comparing the corresponding motif sequences, or structural sequences. In this paper, we measure the performance of such alphabets in the context of the protein structure classification problem. We consider both local and global structural sequences. Each letter of a local structural sequence corresponds to the best matching fragment to the corresponding local segment of the protein structure. The global structural sequence is designed to generate the best possible complete chain that matches the full protein structure. We use an alphabet of 20 letters, corresponding to a library of 20 motifs or protein fragments of size 4 residues. We show that the global structural sequences approximate well the native structures of proteins, with an average cRMS of 0.69 Å over 2225 test proteins. The approximation is best for all α-proteins, while relatively poorer for all β-proteins. We then test the performance of four different sequence representations of proteins (their native sequence, the sequence of their secondary structure elements, and the local and global structural sequences based on our fragment library) with different classifiers in their ability to classify proteins that belong to five distinct folds of CATH. Without surprise, the primary sequence alone performs poorly as a structure classifier. We show that addition of either secondary structure information or local information from the structural sequence considerably improves the classification accuracy. The two fragment-based sequences perform better than the secondary structure sequence, but not well enough at this stage to be a viable alternative to more computationally intensive methods based on protein structure alignment.

Keywords

Protein structure; Structural alphabet; Structure classification; Protein sequence comparison; Sequence feature space

*Corresponding author. quand.le@ucd.ie (Quan Le), gianluca.pollastri@ucd.ie (Gianluca Pollastri), koehl@cs.ucdavis.edu (Patrice Koehl)..

1 Introduction

There is a clear understanding in biology that all cellular functions are deeply connected to the shape of their molecular actors. This is especially true for proteins, whose functions are directly related to their three dimensional structures.[1-4] In hope of deciphering the rules that define the relationships between structure and functions, large scale experimental projects are performed to provide maps of the genetic information of different organisms, including the human genome[5,6] (mostly in the form of genetic sequences of proteins), to derive as much structural information as possible on the products of these genes, and to relate these structures to the function of the corresponding proteins. These are the different "-omics" projects, genomics, structural genomics[7] and functional genomics,[8] to name a few. While these studies are generating a wealth of information, stored into databases, the key to their success lies in our ability to organize and analyze this information, i.e. in our ability to classify proteins, based on their sequences, structures and/or functions, and to connect these classifications (for reviews see[9,10]). In this paper, we focus on the effort of organizing protein structures.

It is currently easier to detect that two proteins share similar functions based on their structures rather than on their sequences. This was observed as early as in 1960, when Perutz et al.[11] showed that myoglobin and hemoglobin, the first two protein structures to be solved at atomic resolution using X-ray crystallography, have similar structures even though their sequences differ. These two proteins are functionally similar, as they are involved in the storage and the transport of oxygen, respectively. Since then, there has been a continued interest in finding structural similarities between proteins, in the hope of revealing shared functionality that could not be detected by sequence information alone. The result of this interest is the development of systems for classification of protein structures that identify and group proteins sharing the same structure so as to reveal evolutionary relationships. Currently, there are three main protein structure classification schemes, SCOP,[12] CATH,[13] and DALI.[14]

Central to any classification scheme is the concept of similarity, and its quantification. A measure of similarity is required to generate the initial classification of the data, as well as to identify the class to which any new data would belong. Defining a similarity measure for protein structures is a difficult problem, leading to discrepancies between the different classification schemes. Protein structure similarity is most often detected and quantified by a protein structure alignment program. Although an approximate optimal solution of the structural alignment problem exists,[15] it is computationally too expensive to be of practical interest. All methods available to-date are heuristic, and consequently at best suboptimal. Many evaluations of structural alignment methods are available.[16-19] These studies usually conclude that an optimal solution to this problem that is fast and reliable and therefore appropriate for classification still need to be defined. As a consequence, there is a significant interest in developing alternative approaches to protein structure alignment for measuring similarities (for a recent review, see[10]).

Finding the (sub) optimal alignment between two protein structures is a hard problem as the rotation and translation of one of the two structures with respect to the other must be found in addition to the alignment itself. By converse, finding the optimal alignment between two protein sequences is a much easier problem, as it can always be solved using dynamic programming, so long as a satisfactory substitution model is available. If it was possible to translate faithfully a protein structure into a string of letters, protein structure comparison would therefore become much easier. This idea of representing structures as a string of letters is in fact grounded in the observation that recurrent, regular structural motifs exist at all levels of organization of protein structures. This was first observed by Corey and Pauling[20,21] and later refined into the concept of protein secondary structures. Although the latter can be predicted

with high accuracy (> 80%), the description of a protein in terms of its secondary structures is not sufficient to capture accurately its three-dimensional geometry. To overcome this limitation, several studies have focused on defining libraries of fragment representatives from which complete protein structures can be modeled with adequate accuracy.[22-29] In these approaches, protein structures are represented as a series of overlapping fragments, each labeled with a symbol, defining a structural alphabet for proteins. The fragments are chosen such as to provide either the best local fits to segments of the protein structure, or the best global fit to the entire protein structure, resulting in two types of structural sequences, namely local or global.[27] Current applications of these structural alphabets include protein structure modeling and in particular decoy generation,[30] local structure prediction,[24,31-34] the reconstruction of a full-atom representation of the protein from the knowledge of the positions of its Cα only,[35,36] identification of structural motifs,[37,38] analysis of protein-protein interactions,[39] as well as protein structure comparison and protein structure database search[29,40-46] We are interested in an extension of the latter, i.e. to the application of structural alphabets to the problem of protein structure classification.

In this paper, we focus on the information content of sequences of proteins, in the context of structure classification (fold classification). More specifically, we compare the performance of five different classifiers, each tested with four different sequence representation of proteins: the native amino acid sequences (NS), the secondary structure element sequence (SSES), and two fragment-based structural sequences, namely a local (LFS) and a global (GFS) sequence of fragments derived from a library of 20 fragments of size 4 residues. We show that LFS, GFS and SSES always outperform the native sequences NS and that GFS and LFS perform statistically significantly better than the SSES when adopted in combination with kernel-based, SVM-based and HMM-based classifiers.

2 Results

With the number of known protein structures in the Protein DataBank[47] growing exponentially, the need for reliable, automatic structure comparison and structure classification tools has never been greater. Here, we test an alternative approach to standard structure comparison in which we use a 1D representation of a protein, or "sequence", to compare and classify proteins. We use four different types of 1D representation: the native sequence of amino acids of the protein (NS), the string describing the secondary structure types of each residue (SSES) and two fragment-based structural sequences, local (LFS) and global (GFS). Figure 1 shows an example of each type of these "sequences" for a small toy protein containing one strand and one helix. We explore two approaches for comparing and classifying sequences. In the first approach, sequences are mapped and compared in a feature space, using a distance measure. In the second approach, sequences are used directly to train either hidden Markov models (HMMs) or support vector machines (SVM) that are subsequently used for classification purpose.

The CATH2225 set of proteins is used as the protein test set for our procedure to generate structural sequences. It includes 2225 proteins grouped into 619 CATH fold classes and was designed such that the sequences of any pair of proteins in the set have statistically no similarity (FASTA[48] E-value > 10^{-4}).[19] We use five of the most populated folds in CATH2225 as a test set for our classifier, covering the three main classes of CATH: one fully α fold (arc repressor mutant), 1.10.10, one fully β fold (immunoglobulin fold), 2.60.40, and three $\alpha - \beta$ folds (TIM fold, 3.20.20, an $\alpha - \beta$ plait, 3.30.70, and the Rossmann fold, 3.40.50). The five folds include a total of 605 proteins from CATH2225; we refer to this set as CATH605. Figure 2 shows examples of protein structures for each of the five folds.

2.1 A 20-letter structural alphabet

In our implementation, we use a library of twenty fragments, each of size four residues. This library was constructed using the approach described by Kolodny and coworkers.[27] The twenty fragments, labeled [A-T], are shown on Figure 3. As expected, these fragments cover all types of local structures for proteins: helices (fragments A, B, C, D and I), strands (F, K, N, R, S) and turns (J). All the other fragments have mixed conformations, and correspond to entrance/exit of helices and strands, as well as loop regions. We have built a tree to represent the similarities between the fragments using the two programs Fitch and Drawtree of the Phylip package.[49] Fitch implements the Fitch and Margulias[50] method for constructing trees from a distance matrix under the "additive tree model" according to which the distance between two objects (fragments) is expected to equal the sum of branch lengths between the objects on the tree. The output of Drawtree is shown on Figure 3; this tree clearly shows the mapping of the fragments to the types of local structures of proteins.

2.2 Fragment-based local structural sequences of proteins

The local structural sequence of a protein is the sequence of fragments of length 4 that best match overlapping windows of 4 residues that scan the protein structure, where match refers to a low cRMS distance. Averaging the cRMS of the best fragment over all windows covering the protein gives the local-fit score for that protein. To test the performance of our fragment library, we computed this score for all 2225 proteins of our CATH2225 domain dataset. Reslts are shown in Figure 4. The average local-fit over CATH2225 is 0.22 Å. Kolodny et al[27] studied the quality of local-fit approximation using libraries of different sizes with fragments of different lengths, using a much smaller test set of proteins (145 proteins). They did not include a library equivalent to the one used here in their study. They derived a relationship between the expected cRMS and the complexity of the library. Their predicted cRMS for a library of 20 fragments of size 4 residues is 0.25, in full agreement with the value observed here. We also calculated the average cRMS deviation for each of the twenty fragments of the library. Figure 5 shows that the helical fragments fit better locally than all other fragments. This makes intuitive sense: helices in proteins are more regular than β-strands, that show greater geometric diversities.[1]

2.3 Fragment-based global structural sequences of proteins

To derive the global structural sequence, we implemented a soft greedy algorithm in which fragments are concatenated without degrees of freedom (similar to stacking lego pieces), such that the coordinate root mean square (cRMS) deviation of the reconstructed structure compared to the native structure is minimal (see the Methods section for more details). To test the performance of both our fragment library and our global-fit algorithm, we reconstructed the 2225 proteins of our CATH2225 domain dataset (see Methods). Results are shown in Figure 6. The average cRMS deviation of the global-fit approximations over the whole dataset is 0.691 Å. Kolodny et al[27] predicted that the average cRMS of global-fit reconstruction for a library of 20 fragments of size 4 residues is 0.70, in full agreement with the value observed here. The best approximation (0.19 Å) is obtained for the domain 2bbvF0, a small helical fragment of 13 residues, while the worst approximation (1.07 Å) is obtained for the domain 1qu5A0, a β-barrel domain of 179 residues that contains many long loops as well as a long unstructured region.

We observe a strong effect of the secondary structure content on the accuracy of the global-fit approximation. Figure 7 shows the global fit approximations of two proteins (2a0b00 and 1hce00) of the same size (118 amino acids), but with different types of secondary structure elements. 2a0b00 is an all α-protein while 1hce00 is an all β-protein; their best approximations achieve an overall cRMS of 0.38 Å and 0.99 Å, respectively. Figure 8 shows plots of the accuracies of the global fit approximations for all proteins in CATH2225 *versus* the helix and

strand contents of the proteins. The accuracy of the approximation improves (cRMS decreases) as helix content increases, with a correlation of 0.8. Conversely, the accuracy of the approximation declines as strand content increases. Both results are consistent with the finding that fragments that best fit helical regions in proteins provide better matches on average than extended fragments (see Figure 5).

2.4 Feature spaces for protein sequences

Each protein sequence is embedded in a space representing the substrings of length p (p-mer) it contains. For $p = 1$, the size of the space is the size N of the alphabet on which the sequence is drawn (20 for NS, GFS and LFS, and 3 for SSES). A protein sequence is represented in this space as a N-dimensional feature vector containing the frequency of each letter of the alphabet in the sequence. A fold in the same space is represented by the mean of the feature vectors of all proteins it contains. To illustrate this process, figure 9 shows the feature vectors in the 1-mer space for all five folds included in CATH605 based on the global structural sequences GFS of their representatives. As expected, the fold 1.10.10 (α-proteins) contains predominantly the fragments A, B, C and I that have been identified as helical fragments (see Figure 3), while the fold 2.60.40 (β-proteins) contains predominantly the fragments F, K, N, O, R and S that correspond to extended regions in proteins. Note that the fragment H is highly represented in these two different folds. Fragment H is neither helical nor extended; it is expected to be found in loops and therefore will appear in the global structural sequences of many types of proteins. The three mixed $\alpha - \beta$ folds contain a mixture of these two sets of fragments. The differential fragment usage based on the secondary structure content of the protein was already observed by Friedberg and colleagues.[44]

The 2–mer space has size N^2: it contains all possible substrings of size 2 that can be formed with the letters of the alphabet for the sequence. The 2–mer space measures the usage of each of these letters, as well as captures their local correlations; as such, it is expected that it provides a better mapping of the sequence space than the 1–mer space, as the conformations of consecutive residues along the backbone of proteins are correlated to one another. In Figure 10, we show two-dimensional representations of the 1–mer and 2–mer feature spaces for all four types of sequences (NS, SSES, GFS and LFS). The 2D representation is created by using metric multidimensional scaling (MDS).[51] We evaluate the clustering of the five folds in the 2D MDS representation using the Average Intercluster Separation, AIS, defined such that a good cluster configuration corresponds to a large AIS value (see methods). There is very little separation between the five folds in the 1–mer space based on the native sequence (AIS=0.26): this is not surprising as the sequences in CATH605 have very little similarity by design. The corresponding 1–mer spaces based on the structural sequences (SSES, GFS and LFS) show better separation of the five folds (AIS=0.58, 0.5 and 0.54, respectively). The all-α (black points) and all-β folds (red points) are well separated; for all three alphabets (SSE, GF and LF) however, the three mixed $\alpha - \beta$ folds overlap significantly. Mapping the native sequences on the 2–mer space improves the representation of the folds; the mixed $\alpha - \beta$ fold 3.40.50 (green points) however still overlap significantly with the α and β folds. In the 2–mer space based on the secondary structure elements, the two mixed $\alpha - \beta$ folds 3.20.20 and 3.30.70 (blue points and magenta points, respectively) are nearly indistinguishable. The 2–mer spaces based on the local and global structural sequences show better separation of the five folds with AIS values of 0.70 and 0.69, respectively, compared to 0.63 for the 2– mer space based on secondary structure and 0.47 for the 2– mer space based on native sequences. The same results were observed for higher dimensional representations of the feature spaces (results not shown).

2.5 Structure similarity versus sequence similarity

Two proteins with highly similar sequences almost always share the same fold. The reverse, however, is not always true: Rost[52] has shown that pairs of proteins with similar structures

possess, on average only 8-10 % sequence identity: this observation is one of the main reasons that it is difficult to classify protein structures based on sequence only. The global structural sequence (GFS) of a protein is designed to capture the characteristics of its structure: a measure of structural sequence similarity should therefore correlate well with a measure of structural similarity. To test this hypothesis, we selected 10,000 diverse pairs of proteins in CATH2225, from highly similar to divergent in structures. For each pair, we computed three different similarities between the two proteins: the L1-norm distances based on the 2-mer feature space for the native sequence (NS) and global structural sequence (GFS), and the structural similarities computed using the structural alignment program STRUCTAL.[53] Figure 11 shows how these measures compare. Native sequence similarity does not correlate well with structure similarity (correlation coefficient: 0.5); the global structural sequences however correlate much better with the structure similarity, with a correlation coefficient of 0.70.

2.6 ROC analysis of protein homology detection

The 2D-representations of the different sequence feature spaces considered above all have in common that similar (i.e. homologous) proteins are mapped to localized regions. To further quantify this observation, we evaluate the performance of similarity measures based on these feature spaces using receiver operating characteristic (ROC) analysis.[54] The five folds in CATH605 serve as the standard: a pair of structures is defined as similar, or “positive”, if they belong to the same fold, and “negative” otherwise. All pairs of proteins in CATH605 are then compared using a similarity measure. For varying thresholds of the measure, all pairs below the threshold are assumed positive, and all above it are negative. The pairs that agree with the standard are called true positives (TP), while those that do not are false positives (FP). ROC analysis compares the rate of TP as a function of the rate of FP; it is scored with the area below the corresponding curve. A ROC score of 1 indicates that all TP are detected first: this corresponds to the ideal measure. On the other hand, a ROC score of 0.5 corresponds to the first diagonal: TP and FP appear at the same rate, and the measure is not discriminative.

We evaluate the performance of the L1-norm distance on the 1-mer and 2-mer feature space of all three sequence representations (native, SSE and fragment-based), and compare with the performance of FASTA[48] and STRUCTAL.[53] Results are shown in Figure 12 and table 2.

FASTA implements a fast Smith and Waterman sequence comparison; the similarity is given either as a raw score, or as an *E*-value; we use the latter as a similarity measure. The ROC curve for the FASTA measure is marginally above the first diagonal, with a score (area) of 0.57: this is expected, as by construction all protein pairs in CATH605 have little or no sequence similarity.

STRUCTAL searches for an optimal alignment of two protein structures using an trial-and-error approach in which an initial alignment is assumed and subsequently refined using dynamic programming. The best alignment is the one with a maximal STRUCTAL score that accounts for the Euclidean distance between the superposed structures and the number of gaps in the alignment.[53] The ROC curve shows that STRUCTAL performs well, with a score of 0.91. This is in agreement with a recent study that compares different structural alignment programs.[19]

The similarity measures based on the L1-norm distance in the feature spaces corresponding to the four sequence representations perform at levels intermediate to FASTA and STRUCTAL. The NS sequence space is closer to FASTA, as it is based on the native sequence of amino acid. The structural sequence spaces SSES, GFS and LFS perform similarly, with ROC scores of 0.85, 0.82 and 0.82, respectively. None perform as well as STRUCTAL: this shows that either the structural sequence or its representation in feature space do not capture all structural features of the proteins.

The L1-norm is only one option for computing the distance between two sequences in their feature space: another approach is to use a kernel to estimate inner product, from which a distance can be derived (see for example[55] and the Method section). We compared the L1-norm ROC scores obtained with a kernel based distance (see table 2) and found very little differences.

2.7 Detecting fold membership

2.7.1 Distance-based classification—The ROC analysis detects protein similarity. We extended the analysis of our sequence similarity measures to the problem of detecting fold membership by performing a set of computational fold classification experiments. In each experiment, we randomly divide the sets of proteins for all five folds that form CATH605 into two groups of approximately equal size: the first groups serve as training sets to define the folds, while the second groups serve as test sets. The test proteins are classified in one of the five folds and the results are stored in a confusion matrix (element (*i*,*j*) of this matrix shows how many proteins belonging to fold *i* are classified as belonging to fold *j*). The accuracy of the classifier is then defined as the ratio of the trace of the confusion matrix over the sum of all its elements (i.e. the percentage of correctly classified proteins). To remove possible bias from the initial separation of the protein set into test and training sets, the procedure is repeated 1000 times. We performed these experiments for the four types of sequences, for two different feature spaces (1-mer and 2-mer), and for two distance measures in these features spaces (namely L1-norm distance and kernel-based distance). The results are reported in table 3. As expected from the feature space representations show above, classifications based on 2-mer representation of protein sequences outperform classifications based on 1-mer features. Interestingly, the kernel-based distances perform better than the L1-norm for the native sequences and the fragment-based structural sequences, but not for the SSE-based structural sequences. This is probably related to the differences in the sizes of the corresponding feature spaces, as NS, GFS and LFS are based on a 20-letter alphabet, while SSE only includes 3 categories (H, E and C). We find that classifications based on structural sequences (SSES, GFS and LFS) always outperform the classifications based on the native sequence only. For completeness, we performed the same experiments using STRUCTAL as a classifier; its success rate is 97.7 ± 0.8 %, confirming that it outperforms our sequence-based classifiers.

2.7.2 HMM-based classification—Table 3 shows that the fragment-based structural sequences perform better than the secondary structure element sequences for fold recognition in protein sequence feature spaces. It is not clear however if this is related to the quality of their representations as high dimensional vectors, or a consequence of the difference of the information content of the sequences themselves. To further characterize the latter, we repeated the fold classification experiments using Hidden Markov Models. For each experiment, we build HMMs on each of the training set for the five folds of CATH605; each protein in the test set is then assigned to one of the five folds, with the HMM model of this fold scoring the highest. The protein sequences representing the fold are considered as the observables, and the hidden states are unknown. We generated a separate model for each of these sequences using the Baum-Welch algorithm;[56] these models were subsequently combined using a simple unweighted average scheme. We fixed the number *N* of hidden states to 4 (see methods for details). The classification accuracies for the HMMs based on the NS, SSES, GFS and LFS are 42.9 %, 63.3 %, 71.5 % and 71.0 %, respectively. As expected, the native sequence alone performs poorly. Interestingly, the HMMs based on the two types of fragment-based structural sequences (global and local) outperform the HMMs based on secondary structure information.

It should be noted that the first order Markovian assumption used in HMMs is clearly violated by fragment-based global structural sequences, as the buildup procedure positions one element

of the sequence based on its three previous elements; as such, the HMMs results should be considered as qualitative.

2.7.3 SVM-based classification—Qiu et al[57] have shown that SVM kernel classifiers perform well for protein structure classification, when combined with structure comparison data. Furthermore, they found that classification based on the structural data alone did not perform as well as when they added the SVM kernel classifier. Leslie et al[55] observed the same behaviour when comparing classification based on native sequence alone, and classification using native sequence and an SVM classifier based on p-mer string kernels Here We tested if these results generalized on different types of protein sequences. For this purpose, we have adapted the multiclass SVM classifier available in Shogun,[58] an open-source machine learning toolbox, to work on our p-mer based string kernel. We tested the classification power of the resulting SVM as follows. The sequences of each fold in Cath605 are randomly divided into two groups, training all testing. The training sequences of the five folds are used to parametrize the multiclass SVM, which is subsequently used to classify the testing sequences. The procedure is repeated 1,000 times, for each type of sequence representation. Results are given in table 4. As expected from the results of Leslie and colleagues,[55] the SVM kernel classifier performs better for fold recognition than the much simpler distance-based classifier, with an average of 10 % increase in performance. The ranking of the different sequence representations however remains identical: the native sequences perform poorly while the two fragment-based structural sequences perform best, with small but statistically significant improvement compared to the sequences based on secondary structure elements.

Table 4 reports the results of experiments using the 1-mer and 2-mer spectrum kernels. We repeated these experiments with higher order spectrum kernels. The performances of the NS, GFS and LFS decreased quickly as the order of the spectrum kernel was increased from 3-mer upward, while the performance of the SSE sequences increased slowly as the order *p*-mer was increased, peaked for a 7-mer kernel with the accuracy 77.1 ± 1.3% and then decreased. These results are just an illustration of the effects of using small data sets and large size alphabets.

3 Discussion

The goal of our study is double: find an appropriate 1D representation of a protein that captures its structural characteristics, and derive a method that uses this 1D representation to classify protein structures.

3.1 Building 1D structural sequences for proteins

To reach our first goal, we have derived a structural alphabet of 20 fragments of size 4 residues using the method proposed by Kolodny et al.[27] We derive two types of 1D structural sequence for a protein using either a local-fit procedure, yielding the local fragment-based sequence (LFS), or a global-fit procedure that builds a chain using these fragments such as to minimize the cRMS between the chain and the actual structure, yielding the global fragment-based sequence, (GFS). We have shown that the LFS matches gives good local fit to the corresponding protein structure, with an average cRMS of 0.25 Å over all its fragments of size 4. The LFS however does not capture the overall 3D geometry of the protein. We have then shown that using the global-fit procedure, we can rebuild proteins with an average accuracy of 0.69 Å, where the average is taken over 2225 proteins (Figure 4). This result cannot be compared directly with other related studies, as the testing sets are usually different. To avoid ambiguity, we repeated our reconstruction experiment on the Park and Levitt set,[59] which has been considered before. We achieved an average accuracy of 0.73±0.13 Å cRMS for the resulting backbone reconstruction. On the same protein set, Camproux et al[28] obtained an average accuracy of 0.64 using a library of 27 fragments of four residues. Kolodny et al[27] observed

that the global-fit accuracy of chains built from structural sequences derived from library of C fragments of length 4 satisfies the empirical equation $< cRMS >= 5.75C^{-0.7}$. Applying this equation to our library and to the library of Camproux et al yields predicted accuracies of 0.71 and 0.57, respectively, well in range with the observed values. Recently, Baeten and colleagues[36] achieved an average reconstruction accuracy of 0.48 Å on the same dataset. In order to reach this accuracy, they used a library of more than 1000 fragments with 6 different lengths. Figure 8 gives us information on why our library did not perform as well as this larger library. We observed that the accuracy with which global structural sequences capture the 3D geometry of a protein improves as the helical content of the protein increases; conversely the accuracy decreases as the strand content of the protein increases. Both observations agree well with the known fact that β-strands have a greater tolerance for deviation from regularity than helices.[1] Strand-like fragments are not under-represented in our library, and in fact make up 1/3 of all fragments (see Figure 3); this is still not large enough to cover the wide basin of conformations observed for residues in strands. Baeten and colleagues overcame this problem by including a very large number of fragments.[36] The question remains as to whether it is possible to reach such accuracy with a much smaller library.

3.2 Structural alphabets and protein structure classification

Protein structure alignment is the most natural method for comparing and/or classifying protein structures. Unfortunately, current algorithms for aligning protein structures are heuristic, and as such cannot guarantee that they find the optimal solution.[19] In addition, the measure of similarity usually reported, i.e. the cRMS, is not a metric for comparing proteins of different sizes, and as such reduces the options in choosing a classifier. An alternative approach to structure alignment is to represent protein structures as vectors in a high dimensional feature space; protein similarity is then quantified in this feature space. In this study, we converted the structure information of a protein into a 1D sequence and then used representation of these 1D sequences in a feature space built upon their substrings to classify protein structures.

We use four different types of sequences: the baseline is the native amino acid sequence and we include three structure-based sequences: a sequence based on secondary structure, and two structural sequences based on fragments, one obtained using local fit to the structure, the other providing the best global-fit to its overall 3D shape.

Native sequences have been used extensively for predicting the structural class or the fold of a protein (see for example[60,61] and references therein). Most of these studies focus on developing new representations of the native sequence; those include histograms of its amino acid composition (which is equivalent to the 1-mer representation we consider),[62,63] pseudo histograms that maintain some information about the order of the amino acids in the sequence,[61] as well as vectors in a feature space indexed by all the sequence p-mers.[55,60,64,65] In the latter studies, Leslie and co-authors project the native protein sequences in a feature space with $k = 3$, 4 or 5 and compare the corresponding vectors using a kernel-based distance and a support vector machine (SVM) classifier. The ROC scores they report for homology detection experiments on a dataset similar to ours are of the order of 0.62 for fold recognition;[55] our ROC scores vary from 0.60 to 0.65, depending on the feature space we use (1-mer or 2-mer), and the distance we use in this feature space (L1-norm or kernel-based). Note that we only use feature spaces indexed by substrings up to 2–*mer* as our training sets are smaller than those used by Leslie et al; we tested $p = 3$, and the ROC scores dropped to 0.48. The loss of accuracy as we increase the word (p-mer) size was also observed by Lingner and Meinicke.[66] The latter authors describe applications of word correlation matrices for remote homology detection, with the words being similar to the p-mer strings we use. They report much better results than those of Leslie at al, or those reported here, with ROC scores in the range 0.8 to 0.92. However, we cannot directly compare however the results of Lingner and Meinicke with ours, as their test

set is much less stringent than ours in the choice of sequences with very little homology. It should be noted that we did not attempt to improve our results on the native sequences, as those were used solely for comparison with the results based on structural sequences.

The local structural sequence LFS of a protein is conceptually equivalent to its sequence of secondary structure elements SSES, with a larger alphabet; both describe well the local structure of a protein, but do not capture its three-dimensional shape. We have shown that local structural sequences induce better fold classification than secondary structure sequences, based on distance-based, HMM-based and SVM-based classifiers. The improvement however, though significant, is not commensurate with the difference of the richness of the alphabets (from 3 letters for SSEs to 20 letters for LFS) This makes intuitive sense as helical and extended (strand) conformations dominate among local conformations of proteins, as observed for example on the Ramachandran plot describing a protein. In parallel, we have shown the global structural sequences GFS perform with the same success level as local structural sequences. This result is more surprising. Intrinsically, a GFS contains more information than the SSES or LFS, as it captures the protein's 3D shape: we have shown for example that we can reconstruct the full 3D structure using the former within 0.7 Å, i.e. with a high accuracy level. Our results indicate that the three types of classifiers we have used, namely distance-based, HMM-based and SVM-based, are not able to capture this information. The reasons for this failure are unclear at this stage: the small size of the *p*-mer (limited to 2 as data become scarce for larger strings), the quality of the kernel (basically linear in our case), the choice of hidden states for HMM are all possible causes that require further investagation. For example, the HMM we have built do not take into account position, as we did not align the set of sequences representing a fold (this is usually done in HMM-based fold recognition techniques). While it is relatively easy to align multiple native sequences of proteins using dynamic programming and a mutation matrix, it is more difficult to align structural sequences, because of the strong correlations between neighboring fragments. Recent developments in this area[29,40-46] give us hope that we will be able to develop more performant HMMs based on structural sequences.

Structural sequences have already been used for structure classification. Wang et al used HMM learning on sequences of local structural alphabets (LSA) to classify protein structure folds; [41] they report accuracies of fold classification of 82 % on training-set structures sharing less than 40 % pairwise sequence identity and 65 % for those sharing no more than 25 % sequence identity. Our results on the local structural sequences LFS fall between these two values, with accuracies of the order of 70 %. Wang et al also tested structural alphabets of different sizes, and showed that results improve as the alphabet size increases up to 20 letters but then reaches a plateau. This comforts us in our choice to maintain a small alphabet size to derive structural sequences.

3.3 How do we compare structural sequences?

While pattern recognition techniques attract a lot of interest for comparing protein sequences, sequence alignment using dynamic programming remains the method of choice when comparing native sequences of proteins. There are however at least two problems when we try to apply sequence alignment to structural sequences. Firstly, we need a substitution matrix and values for the gap penalties. There has been several attempts to derive such values for structural alphabets;[29,44] alignments using these parameters are not yet equivalent to the corresponding direct structural alignments. Secondly, special care is needed to make sure that dynamic programming applies on these sequences; problems occur for global structural sequences, as the letters in the sequence are not independent of each other. In this study, we avoided these issues by projecting the sequence in a feature space, and using pattern recognition techniques to compare sequences in this space. We have shown however that these techniques do not

extract all the information contained in the structural sequences. We expect that success will come from first solving the structural sequence alignment problem.

Conclusion

Structural biology recently experienced a major uplift through the development of high throughput studies aimed at developing a comprehensive view of the protein structure universe. The key to the success of this approach lies in our ability to organize and analyze the wealth of information it generates and to integrate that information with other efforts in cellular biology. Protein structure comparison and classification are obligatory steps of this process and as such are the focus of many research studies (for review, see[10]). Protein structure alignment is probably the most natural approach to performing these two steps. However, difficulties in generating accurate structural alignments[19,67] and ambiguities of the RMSD, the standard measure of structure similarity (which is not a metric when comparing proteins of different sizes) is pushing the field of computational biology to search for alternative approaches that may alleviate these problems. In this study, we explored the option to encode the 3D structure of a protein into a 1D string or sequence and to use standard pattern recognition techniques to compare and classify these sequences. We have shown that both local and global 1D representations based on a structural alphabet of sequential protein fragments map the protein structure accurately, perform significantly better than the 1D representation based on the secondary structure content of the protein for fold recognition purposes, but not enough to become a viable replacement to computationally intensive procedures such as structural alignment tools. We plan to extend this study in several directions. Firstly, we will analyze in greater detail the reasons of this apparent failure. In particular, we will test more sophisticated sequence learning tools to capture long range correlations from sequence data. Then, we want to take into account the fact that the 1D global structural sequence of a protein is not unique; by using profiles of 1D structural sequences, we expect to be able to exploit more sophisticated sequence analysis methods to compare protein structures. Secondly, we plan to extend beyond fold recognition and use our structural sequences for generating protein structural alignment, along the lines of recent work of Friedberg et al.[44] Another direction is to study the correlations between the different sequence representations of a protein. The idea that it might be possible to predict the structural sequence of a protein based on its native sequence is very appealing.

4 Methods

4.1 Protein Fragment Library

We represent protein structures using a library of 20 fragments of protein backbone. Each fragment is 4 residue long and is defined by the three-dimensional coordinates of its 4 Cα atoms. The library was generated from 200 protein domains whose structure was accurately determined by X-ray crystallography. Each of these domains was broken into a series of non overlapping fragments of 4 residues, which were consequently grouped into 20 clusters, according to their cRMS deviation from one another. Each cluster is represented by a single element -the cluster's center. This work has been described in detail elsewhere.[27] Note that the fragments do not include any sequence specifics of the proteins.

4.2 Structural Sequences

The 20 fragments in our library serve as building blocks for constructing 1D representations of protein structures. Two types of 1D sequences, are considered, designed to capture local or global structural features of the protein, respectively.

The best local-fit representation of a protein is derived in linear-time by finding for each fragment of four residues of the protein the most similar fragment in the library, where similarity is measured using coordinate RMS.

We build the global structural sequence of a protein using the method introduced by Kolodny and co-workers.[27] Namely, copies of fragments from the library are repeatedly added to extend a chain, until reaching the target length, under the constraint that the overall cRMS distance between the model and the native structure of the protein is minimal. Each new fragment is added by superimposing its first three atoms on the last three atoms of the growing chain, extending the chain by 4 – 3 = 1 atom (where "atoms" refers to the Cα of each amino acid of the protein). An important property of this method is that each step of adding a fragment is local, while the measure of goodness of fit is global. Once the full chain is constructed, the best approximation forms the model for the structure. Each fragment is then assigned a single letter label (from A to T for the 20 fragments), and the final model for a protein structure can be stored as a linear string of these labels with negligible loss of information.

The number of possible global-fit approximations for a protein of N residues with our library of 20 fragments of size 4 is 20^{N-3}; this number is too large to be explored systematically. Our implementation is consequently heuristic, and uses a soft greedy algorithm with a heap *Nkeep*. The choice of *Nkeep* is a balance between the desire to reach a high accuracy and the reality of the computing time and memory usage. To decide on which value of *Nkeep* to choose, We tested the quality of the global fit approximations of 2225 proteins (see below) for 5 values of *Nkeep*. Results are shown in table 1. As expected, higher values of *Nkeep* lead to better approximations. The accuracy however improves drastically with increasing *Nkeep* for small heap sizes, and remains roughly constant for larger heap sizes. We consequently used a heap size of 4000 for all our studies. We also find that the CPU time required to compute one approximation depends roughly linearly on *Nkeep* (see table 1).

It is important to notice that the local structural sequence of a protein is unique, while usually many global structural sequences represent the 3D protein structures with similar accuracies. In this study, we define the global structural sequence of a protein as the highest scoring sequence derived by our algorithm. As the latter is heuristic, there are no guarantees that this sequence is indeed the best sequence that can be constructed. To ensure that this approximation had no impact on the results presented here, we repeated all classification experiments with a new definition of the global structural sequence, chosen at random among the top 50 best scoring sequences obtained using the global-fit procedure. Results remained the same, both qualitatively and quantitatively.

4.3 Data set of protein structures

The set of structures considered in this study is extracted from the database of 2930 sequence-diverse CATH v2.4 domains used in a previous study.[19] As we focus on three-dimensional structures, we consider the first three levels of CATH, Class, Architecture and Topology, to give a CAT classification. We refer to a set of structures with the same CAT classification as a *fold class*. Using a set of structures with sufficient sequence diversity ensures that the data is duplicate-free and that the problem of detecting structural similarity is non-trivial for all pairs of proteins considered. The 2930 structures were selected as follows: (i) sort all 34,287 CATH v2.4 domains by their SPACI score.[68] (ii) Start with the domain with highest SPACI score, and remove from the list all domains that share significant sequence similarity with it (FASTA E-value < 10^{-4}. (iii) Repeat step (ii) with all domains in the list that have not been removed. This is the same procedure used by Brenner et al. to generate the sequence-diverse set of SCOP structures. The set of 2930 resulting from this procedure was further filtered to remove all proteins whose experimental structures have gaps in their backbone. The final set contains 2225 proteins, and is referred to as CATH2225.

There are 619 fold classes in CATH2225, many of which only contain a single element (394). To facilitate statistical analysis, we selected five of the most populated folds in CATH2225 as the test set for all computational experiments run in these studies, including at least one fold from each CATH class: CATH fold 1.10.10, a fully α fold (arc repressor, 62 representatives), CATH fold 2.60.40, a fully β fold (immunoglobulin-like, 169 representatives), and three mixed $\alpha - \beta$ folds: 3.20.20, (TIM-like, 67 representatives), 3.30.70, (two layer sandwich, 92 representatives) and 3.40.50 (Rossmann fold, 215 representatives). These five folds include a total of 605 proteins of CATH2225 (set CATH605). Figure 2 shows examples of protein structures for each of these five folds.

4.4 Classifiers

Our goal is to define protein structure classifiers based on sequence information (where sequence can be the native amino acid sequence of the protein, or a structural sequence representing the structure). The number of known protein structures is growing exponentially, and the number of folds representing this structures is still growing: scalability is therefore an essential feature that we take into account when designing of our classifiers. As a consequence, we opt for relatively simple classification models.

The dynamics of the sequences are considered in the form of statistics of n-gram subsequences of different string representation. In the following subsections we describe the three different types of classifiers we adopt in this work: distance based (nearest neighbor) discriminative models, generative models in the form of Hidden Markov Models (HMMs) and the multiclass Support Vector Machines (SVMs).

4.4.1 Distance based methods—In the distance based methods, each protein sequence is embedded in a high dimensional vector space in such a way that the distance between two proteins in this space reffects their similarity. Here “sequence” refers to either the native sequence (NS) of amino acids of the protein, the structural sequence obtained from the secondary structure elements (SSES) of the protein, or through reconstruction using a fragment library (FS). The pairwise distances are then used to classify the proteins. The different approaches used here reduce to various ways of counting sub-strings that two strings have in common. This is a meaningful similarity notion in biological applications since evolutionary proximity is thought to result both in structure similarity (functional similarity) and in sequence similarity. A distance-based classifier is fully characterized by: (a) the mapping of a sequence in a feature space, (b), the representation of a fold class in this feature space, and (c), the classification procedure itself.

The embedding map is defined between the space of all finite sequences over an alphabet Σ (amino acids for primary sequences, or structural units for structural sequences) and a vector space F. The coordinates of F are indexed by a subset I of strings over Σ, that is by a subset of the input space. In this paper we will study embeddings where I is the set Σ^p of strings of length p. In practice, we will limit ourselves to the case $p = 2$, but the theory remains valid for all p. We use $\bar{\phi}$ to denote the feature mapping

$$\bar{\phi} : s \in \Sigma^* \mapsto \left(\bar{\phi}_u(s)\right)_{u \in I} \in F. \tag{1}$$

The embedding of protein sequences to the p-mer subsequence space is done either explicitly by computing the p-mer probability distribution of each protein sequence, or by using implicit embedding with p-spectrum kernel,[69,70] the latter leading to much faster implementation (i.e. with a linear complexity). We report here both of these methods.

a. L1-norm distance method of p-gram probability distribution In this method, we compute explicitly the probability distribution of *p*-mer for each protein. Each fold can be represented by the mean of all proteins belonging to it, and the distance between two protein sequences in the feature space can be computed as follows

$$d(s,t) = \sum_{u \in \Sigma^p} |\bar{\phi}^p_u(s) - \bar{\phi}^p_u(t)|. \tag{2}$$

b. ***p*-spectrum kernel** Perhaps the most natural way to compare two strings is to count how many (contiguous) substrings of length *p* they have in common. We define the *spectrum of order p* (or *p-spectrum*) of a sequence *s* to be the histogram of frequencies of all its (contiguous) substrings of length *p* (*p*-mer). We define the *p*-spectrum kernel as the inner product of the *p*-spectra. Formally, the feature space *F* associated with the *p*-spectrum kernel is indexed by $I = \Sigma^p$, with the embedding given by

$$\phi^p_u(s) = |\{(v_1, v_2) : s = v_1 u v_2\}|, u \in \Sigma^p \tag{3}$$

and the associated *p*-spectrum kernel between sequences *s* and *t* is defined as

$$\kappa_p(s,t) = \langle \phi^p(s), \phi^p(t) \rangle = \sum_{u \in \Sigma^p} \phi^p_u(s)\, \phi^p_u(t). \tag{4}$$

To turn the *p*-spectrum kernel into the inner product of *p*-mer probability distribution feature space, we need to normalize the feature vectors (in other words, this step removes the effect of the length of each sequence). This can be done by the following operation:

$$\bar{\kappa}_p(s,t) = \frac{\kappa_p(s,t)}{\sqrt{\kappa_p(s,s)\,\kappa_p(t,t)}}. \tag{5}$$

The kernel functions represent inner products in a feature space. Given the kernel values, the distance between the feature vectors corresponding to two sequences *s* and *t* can be computed as

$$d(s,t)^2 = \bar{\kappa}_p(s,s) + \bar{\kappa}_p(t,t) - 2\bar{\kappa}_p(s,t) = 2 - 2\bar{\kappa}_p(s,t). \tag{6}$$

Note that to compute the distance from a protein sequence to a fold, we need the notion of the mean of a fold *S* in the feature space.

$$\bar{\phi}^p_S = \frac{1}{l} \sum_{i=1}^{l} \bar{\phi}^p(s_i) \tag{7}$$

where $\{s_i\}_{i=1}^{l} \in S$ are protein sequences in the fold.

Even though it is impossible to represent explicitly the mean of a fold in the feature space, we can nevertheless compute its norm

$$\|\bar{\phi}_s^p\|_2^2 = \left\langle \bar{\phi}_s^p, \bar{\phi}_s^p \right\rangle = \frac{1}{l^2} \sum_{i,j=1}^{l} \bar{\kappa}_p(s_i, s_j) \tag{8}$$

and the distance between the mean of a fold and the image of a protein sequence is given by

$$\begin{aligned} \|\bar{\phi}^p(s) - \bar{\phi}_s^p\|_2^2 &= \left\langle \bar{\phi}^p(s), \bar{\phi}^p(s) \right\rangle + \left\langle \bar{\phi}_s^p, \bar{\phi}_s^p \right\rangle - 2\left\langle \bar{\phi}_s^p, \bar{\phi}^p(s) \right\rangle \\ &= \bar{\kappa}_p(s,s) + \frac{1}{l^2} \sum_{i,j=1}^{l} \bar{\kappa}_p(s_i, s_j) - \frac{2}{l} \sum_{i=1}^{l} \bar{\kappa}_p(s, s_i) \end{aligned} \tag{9}$$

4.4.2 Hidden Markov Models—Hidden Markov Models (HMMs) are statistical models for modeling sequential data; they have been used successfully in pattern recognition, speech processing and biosequences modeling. Rabiner wrote a good introduction to HMMs.[71] The joint probability of a sequence of observations $y_1^T = y_1, y_2, , ..y_T$ can always be factored as

$$P\left(y_1^T\right) = P(y_1) \prod_{t=2}^{T} P\left(y_t | y_1^{t-1}\right) \tag{10}$$

Its calculation appears intractable in general. However, if we assume that the past sequence can be summarized by a *state variable* q_t, then one can rewrite the previous equation as

$$P\left(y_1^T\right) = \sum_{q_1}^{T} P\left(y_1^T, q_1^T\right) \tag{11}$$

where the sum over q_1^T represents the sum over all possible state sequences $q_1^T = q_1, q_2, ..q_T$ of length T. Fortunately now, this can be factored as follows

$$P\left(y_1^T, q_1^T\right) = P(q_1) \prod_{t=2}^{T} P(q_t | q_{t-1}) \prod_{t=1}^{T} P(y_t | q_t) \tag{12}$$

using the first order Markovian assumption (one state depends probabilistically on just the preceding state).[72] The joint probability is therefore completely specified in terms of *initial state probabilities* $P(q_1)$, *transition probabilities* $P(q_t|q_{t-1})$ and *emission probabilities* $P(y_t|q_t)$. Since each state variable q_t for the underlying Markov model is not directly observed but is a stochastic function of the previous observations y^{t-1}, such a model is called Hidden Markov Model. It can be trained to maximize the likelihood of a training set of sequences (Maximum Likelihood criterion), using well known algorithms such as Expectation Maximization (EM),[73] or Viterbi.[74]

For classification tasks, such as deciding if a given sequence belongs to a given target class (fold in our case), one usually trains a different HMM for each class (fold) to maximize the likelihood that the training sequences are assigned to that class. Then, given a new sequence to classify, one computes the likelihood of the sequence for each HMM, and selects the class that maximizes the likelihood criterion, $P\left(y_1^T | \text{class}=c\right)$.

The HMM of a fold is constructed from a training set of sequences it contains. Individual models are constructed for each training sequence using the Baum-Welch procedure,[56] and subsequently combined using a simple unweighted average scheme to define the parameters of the HMM for the fold. The number of hidden states N is chosen as the value that optimizes the classification power of the HM models. We tested values of N between 1 and 20, and retained the values of 4 as it gave the best results for all four types of sequences. Finally, we compute the probability of a sequence of observations given a model using the forward procedure.[71]

It should be noted that the models based on the two types of fragment-based structural sequences and to a lesser extend the models based on secondary structure elements violate the first order Markovian assumption because each structural element is statistically related to its neighbouring structural elements in the sequence. The lack of data (about one hundred protein are available for each fold) did not allow us to build a more complicated model.

4.4.3 Support Vector Machines—Support vector machines (SVM) belong to the class of supervised learning methods used for classification and regression. They were initially introduced to solve binary classification problems, in which they attempt to separate the data represented as two sets of vectors in a feature space by constructing a hyperplane in that space that maximizes the margin between the two data sets.[75]

Let (x_i, y_i) be a dataset with N examples, where x_i is the *ith* input and label y_i is either 1 or −1, indicating the class to which the point x_i belongs. Let $\phi(x_i)$ be the high dimensional vector representing the point x_i in its feature space. An hyperplane in this feature space can be written as $f(x) = \langle w, \phi(x) \rangle + b = 0$. If this hyperplane divides the points having $y_i = 1$ from those having $y_i = -1$, then $sign(f(x))$ is a classification function for the data. The *weight* vector w (which is the normal to the hyperplane) and the *bias* b are determined by solving an optimization problem over the training data. A key feature of SVM is that in this optimization problem the feature vectors $\Phi(x)$ only appear in dot products; the explicit mapping Φ from the data space to the feature space can then be replaced with an implicit mapping through the use of a kernel function $k(x, x') = \langle \Phi(x), \Phi(x') \rangle$ which does not need to be linear.

There are different extensions of the binary SVMs to solve multiclass learning problems. The three most popular are: pairwise classification (one versus one), one versus all, and a direct approach based on a multi-class objective functions.[76] In our experiments with multiclass SVM we used the pairwise classification method (one versus one) and the spectrum kernel defined in equation (4) to classify the proteins in Cath605. In the one versus one classification method, one trains a binary SVM for each possible pair of classes using examples from those two pairs. For m classes, this results in $\frac{(m-1)m}{2}$ binary SVMs. To classify a test input, all binary classifiers are evaluated and the input is assigned to the class which has the highest number of votes.

We used the multiclass SVM code and adapted our spectrum kernel in the open source package Shogun.[58] For comparison, we implemented also the two other popular methods for multiclass SVM (i.e. the one versus all and the one using the multiclass objective function); the results were very similar.

4.4.4 Visualizing data using Multi Dimensional Scaling—Multi Dimensional Scaling (MDS) is a technique used to provide a low dimensional 'mapping' (usually two or three dimensions) of high-dimensional data points.[51] We apply the standard Metric MDS. Metric MDS starts with a (nxn) distance matrix D, whose element d_{ij} is the Euclidian distance between data point i and data point j. Distance geometry is used to convert this distance matrix into the inner product matrix B, for which element b_{ij} is the inner product of the vectors representing the data point i and data point j in the Euclidean space. The next step computes the first few

principal eigenvectors (usually two or three) from the inner product matrix B and then project all data points on these principal eigenvectors, thereby generating a low dimensional mapping.

The five folds in the CATH605 dataset are represented as five clusters in the low dimensional mapping obtained by MDS. If the features defining the mapping capture well the differences between the folds, these five clusters ought to be well separated. We evaluate this statement using the Average Intercluster Separation (AIS):

$$AIS = \frac{2}{N_c\,(N_c - 1)} \sum_{i=1}^{N_c - 1} \sum_{j=i+1}^{N_c} d\left(C_i, C_j\right) \tag{13}$$

where the distance between two clusters C_i and C_j is defined as:

$$d\left(C_j, C_j\right) = \frac{1}{|C_i||C_j|} \sum_{i \in C_i} \sum_{j \in C_j} d\,(i, j)\,. \tag{14}$$

$d(i, j)$ is the Euclidian distance between the data points i and j in the projected space, and $|C_i|$ is the size of cluster C_i. If two clusters C_i and C_j are well separated, all inter distances between their members are large and the average distance $d(C_i, C_j)$ is consequently large. Therefore larger values for AIS means better cluster configuration.

Acknowledgments

We thank Rachel Kolodny for generating the library of fragments used in this study. Quan Le wishes to thank Nello Cristianini for his supervision and helpful suggestions during Quan's visit in UC Davis. Patrice Koehl acknowledges support from the Alfred P. Sloan foundation and from the NIH. Gianluca Pollastri and Quan Le's work is funded by by Science Foundation Ireland grant 05/RFP/CMS0029 and grant RP/2005/219 from the Health Research Board of Ireland. Quan's visit to UC Davis was partly funded by a 2005 UC Dublin Seed Funding award.

References

1. Richardson J. The anatomy and taxonomy of protein structure. Adv. Protein. Chem 1981;34:167–339. [PubMed: 7020376]
2. Chothia C. Principles that determine the structure of proteins. Annu. Rev. Biochem 1984;53:537–572. [PubMed: 6383199]
3. Branden, C.; Tooze, J. Introduction to protein structures. Garland Publishing; New York: 1991.
4. Creighton, T. Proteins. W.H. Freeman and Co; New York: 1993.
5. International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. Nature 2001;409:860–921. [PubMed: 11237011]
6. Venter J, et al. The sequence of the human genome. Science 2001;291:1304–1351. [PubMed: 11181995]
7. Burley S, Almo S, Bonanno J, Capel M, Chance M, Gaasterland T, Lin D, Sali A, Studler F, Swaminathan S. Structural genomics: behond the human genome project. Nat. Genet 1999;23:151–157. [PubMed: 10508510]
8. Hieter P, Boguski M. Functional genomics: it's all how you read it. Science 1997;278:601–602. [PubMed: 9381168]
9. Taylor W, May A, Brown N, Aszodi A. Protein structure: geometry, topology and classification. Reports Prog. Phys 2001:517–590.
10. Koehl, P. Reviews in Computational Chemistry. Vol. Vol. 22. Wiley; 2006. Protein structure classification; p. 1-48.

11. Perutz M, ROssman M, Cullis A, Muirhead G, Will G, North A. Structure of haemoglobin: a three-dimensional Fourier synthesis at 5.5 Angstrom resolution, obtained by X-ray analysis. Nature 1960;185:416–422. [PubMed: 18990801]
12. Murzin A, Brenner S, Hubbard T, Chothia C. Scop: A structural classification of protein database for the investigation of sequences and structures. J. Mol. Biol 1995;247:536–540. [PubMed: 7723011]
13. Orengo C, Michie A, Jones S, Jones D, Swindells M, Thornton J. CATH: A hierarchic classification of protein domain structures. Structures 1997;5:1093–1108.
14. Holm L, Sander C. Protein structure comparison by alignment of distance matrices. J. Mol. Biol 1993;233:123–138. [PubMed: 8377180]
15. Kolodny R, Linial N. Approximate protein structural alignment in polynomial time. Proc. Natl. Acad. Sci. (USA) 2004;101:12201–12206. [PubMed: 15304646]
16. Sauder J, Arthur J, Dunbrack R. Large scale comparison of protein sequence alignment algorithms with structural alignments. Proteins: Struct. Func. Genet 2000;40:6–22.
17. Sierk M, Pearson W. Sensitivity and selectivity in protein structure comparison. Protein Sci 2004;13:773–785. [PubMed: 14978311]
18. Novotny M, Madsen D, Kleeywegt G. Evaluation of protein fold comparison servers. Proteins: Struct. Func. Genet 2004;54:260–270.
19. Kolodny R, Koehl P, Levitt M. Comprehensive evaluation of protein structure alignment methods: scoring by geometric measures. J. Mol. Biol 2005;346:1173–1188. [PubMed: 15701525]
20. Pauling L, Corey R, Branson H. The structure of proteins: two hydrogen-bonded helical configurations of the polypeptide chain. Proc. Natl. Acad. Sci. (USA) 1951;37:205–234. [PubMed: 14816373]
21. Pauling L, Corey R. Configuration of polypeptide chains with favored orientations around a single bonds: two new pleated sheets. Proc. Natl. Acad. Sci. (USA) 1951;37:729–740. [PubMed: 16578412]
22. Rooman M, Rodriguez J, Wodak S. Automatic definition of recurrent local structure motifs in proteins. J. Mol. Biol 1990;213:327–336. [PubMed: 2342110]
23. Fetrow J, Palumbo M, Berg G. Patterns, structures, and amino acid frequency in structural building blocks, a protein secondary structure classification scheme. Proteins: Struct. Func. Genet 1997;27:249–271.
24. de Brevern A, Etchebest C, Hazout S. Bayesian probabilistic approach for predicting backbone structures in terms of protein blocks. Proteins: Struct. Func. Genet 2001;41:271–287.
25. Bystroff C, Thorsson V, Baker D. HMMSTR: a hidden Markov model for local sequence-structure correlation in proteins. J. Mol. Biol 2000;301:173–190. [PubMed: 10926500]
26. Yang A, Wang L. Local structure-based sequence profile database for local and global protein structure predictions. Bioinformatics 2002;18:1650–1657. [PubMed: 12490450]
27. Kolodny R, Koehl P, Guibas L, Levitt M. Small libraries of protein fragments model native protein structures accurately. J. Mol. Biol 2002;323:297–307. [PubMed: 12381322]
28. Camproux A, Gautier R, Tuffery P. A hidden Markov model derived structural alphabet for proteins. J. Mol. Biol 2004;339:591–605. [PubMed: 15147844]
29. Tung C-H, Huang J-W, Yang J-M. Kappa-alpha plot derived structural alphabet and BLOSUM-like substitution matrix for rapid search of protein structure database. Genome Biol 2007;8:R31.1–R31.16. [PubMed: 17335583]
30. Kolodny R, Levitt M. Protein decoy assembly using short fragments under geometric constraints. Biopolymers 2003;68:278–285. [PubMed: 12601789]
31. Camproux A, de Brevern A, Hazout S, Tuffery P. Exploring the use of a structural alphabet for structural prediction of protein loops. Theor. Chem. Acc 2001;106:28–35.
32. Fourrier L, Benros C, de Brevern A. Use of a structural alphabet for analysis of short loops connecting repetitive structures. BMC Bioinformatics 2004;5:58–71. [PubMed: 15140270]
33. Etchebest C, Benros C, Hazout S, de Brevern A. A structural alphabet for local protein structures: improved prediction methods. Prot. Struct. Func. Bioinf 2005;59:810–827.
34. Mooney C, Vullo A, Pollastri G. Protein structural motif prediction in multidimensional ϕ-ψ space leads to improved secondary structure prediction. J. Comp. Biol 2006;13:1489–1502.

35. Maupetit J, Gautier R, Tuffery P. SABBAC: online structural alphabet-based protein backbone reconstruction from alpha-carbon trace. Nucl. Acids. Res 2006;34:W147–W151. [PubMed: 16844979]
36. Baeten L, Reumers J, Tur V, Stricher F, Lenaerts T, Serrano L, Rousseau F, Schymkowitz J. Reconstruction of protein backbones from the BriX collection of canonical protein fragments. PLOS Comput. Biol 2008;4:e1000083. [PubMed: 18483555]
37. Kurgan L, Kedarisetti K. Sequence representation and prediction of protein secondary structure for structural motifs in twilight zone proteins. The Protein J 2006;25:463–474.
38. Dudev M, Lim C. Discovering structural motifs using a structural alphabet application to magnesium-binding sites. BMC Bioinformatics 2007;8:106–117. [PubMed: 17389049]
39. Martin J, Regad L, Lecornet H, Camproux A-C. Structural deformation upon protein-protein interaction: a structural alphabet approach. BMC Struct. Biol 2008;8:12–30. [PubMed: 18307769]
40. Guyon F, Camproux A, Hochez J, Tuffery P. SA-Search: a web tool for protein structure mining based on a structural alphabet. Nucl. Acids. Res 2004;32:W545–W548. [PubMed: 15215446]
41. Wang S, Chen C, Hwang M. Classification of protein 3d folds by hidden markov learning on sequence of structural alphabets. Proc. 3rd Asia-Pacific Bioinf. Conf. (APBC) 2005:65–72.
42. Tyagi M, Gowri V, Srinivasan N, de Brevern A, Offmann B. A substitution matrix for structural alphabet based on structural alignment of homologous proteins and its applications. Prot. Struct. Func. Bioinf 2006;65:32–39.
43. Yang J, Tung C. Protein structure database search and evolutionary classification. Nucl. Acids. Res 2006;34:3646–3659. [PubMed: 16885238]
44. Friedberg I, Harder T, Kolodny R, Sitbon E, Li Z, Godzik A. Using an alignment of fragment strings for comparing protein structures. Bioinformatics 2006;23:e219–e224. [PubMed: 17237095]
45. Lo W-C, H. P-J, CHang C-H, Lyu P-C. Protein structural similarity search by Ramachandran codes. BMC Bioinformatics 2007;8:307–320. [PubMed: 17716377]
46. Zheng, W-M. The use of a conformational alphabet for fast alignment of protein structures. In: Mandoiu, I.; Sunderraman, R.; Zelikovsky, A., editors. ISBRA 2008. Springer Verlag; Berlin, Germany: 2008. p. 331-342.
47. Berman H, Westbrook J, Feng Z, Gilliland G, Bhat T, Weissig H, Shindyalov J, Bourne P. The protein data bank. Nucl. Acids. Res 2000;28:235–242. [PubMed: 10592235]
48. Pearson W, Lipman D. Improved tools for biological sequence comparison. Proc. Natl. Acad. Sci. (USA) 1988;85:2444–2448. [PubMed: 3162770]
49. Felsenstein J. Phylogeny inference package (version 3.2). Cladistics 1989;5:164–166.
50. Fitch W, Margoliash E. Construction of phylogenetic trees. Science 1967;155:279–284. [PubMed: 5334057]
51. Kruskal, J.; Wish, M. Multidimensional Scaling. Sage publication; Beverly Hills, CA: 1978.
52. Rost B. Protein structures sustain evolutionary drift. Fold. Des 1997;2:519–524.
53. Subbiah S, Laurents D, Levitt M. Structural similarity of dna-binding domains of bacteriophage repressors and the globin fold. Curr. Biol 1993;3:141–148. [PubMed: 15335781]
54. Gribskov M, Robinson N. Use of receiver operating characteristic (ROC) analysis to evaluate sequence matching. Comput. Chem 1996;20:25–33. [PubMed: 16718863]
55. Leslie C, Eskin E, Cohen A, Weston J, Noble W. Mismatch string kernels for discriminative protein classification. Bioinformatics 2004;20:467–476. [PubMed: 14990442]
56. Baum LE, Petrie T, Soules G, Weiss N. A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. Ann. Math. Statist 1970;41:164–171.
57. Qiu J, Hue M, Ben-Hur A, Vert J-P, Noble WS. A structural alignment kernel for protein structures. Bioinformatics 2007;23:1090–1098. [PubMed: 17234638]
58. Sonnenburg S, Raetsch G, Schaefer C, Schoelkopf B. Large scale multiple kernel learning. J. Machine Learning Res 2006;7:1531–1565.
59. Park B, Levitt M. The complexity and accuracy of discrete state models of protein structure. J. Mol. Biol 1995;249:493–507. [PubMed: 7783205]

60. Melvin I, Ie E, Kuang R, Weston J, Noble W, Leslie C. SVM-fold: a tool for discriminative multi-class protein fold and superfamily recognition. BMC Bioinformatics 2007;8:S2. [PubMed: 17570145]
61. Xiao X, Lin W-Z, Chou K-C. Using grey dynamic modeling and pseudo amino acid composition to predict protein structural classes. J. Comp. Chem 2008;29:2018–2024. [PubMed: 18381630]
62. Nakashima H, Nishikawa K, Ooi T. The folding type of a protein is relevant to the amino acid composition. Biochem. J 1986;99:153–162.
63. Chou K. A novel approach to predicting protein structural classes in a (20-1)-D amino acid composition space. Proteins: Struct. Func. Genet 1995;21:319–344.
64. Shamim M, Anwaruddin M, Nagarajaram H. Support Vector Machine-based classification of protein folds using the structural properties of amino acid residues and amino acid residue pairs. Bioinformatics 2007;23:3320–3327. [PubMed: 17989092]
65. Damoulas T, Girolami M. Probabilistic multi-class multi-kernel learning: on protein fold recognition and remote homology detection. Bioinformatics 2008;24:1264–1270. [PubMed: 18378524]
66. Lingner T, Meinicke P. Word correlation matrices for protein sequence analysis and remote homology detection. BMC Bioinformatics 2008;9:259–271. [PubMed: 18522726]
67. Koehl P. Protein structure similarities. Curr. Opin. Struct. Biol 2001;11:348–353. [PubMed: 11406386]
68. Brenner S, Koehl P, Levitt M. The astral compendium for protein structure and sequence analysis. Nucl. Acids. Res 2000;28:254–256. [PubMed: 10592239]
69. Taylor, JS.; Cristianini, N. Kernel Methods for Pattern Analysis. Cambridge University Press; Cambridge: 2004.
70. Lodhi H, Saunders C, Taylor J, Cristianini N, Watkins C. Text classification using string kernels. J. Mach. Learn. Res 2002:419–444.
71. Rabiner LR. A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of the IEEE 1989;77:257–285.
72. Rabiner, L.; Juang, B-H. Fundamentals of speech recognition. Prentice Hall; Upper Saddle River, NJ, USA: 1993.
73. Dempster AP, Laird NM, Rubin DB. Maximum-likelihood from incomplete data via the EM algorithm. J. Royal Stat. Soc. B 1977;39:1–38.
74. Viterbi A. Error bounds for convolutional code and an asymptotically optimum decoding algorithm. IEEE Trans. Inf. Theory 1967:260–269.
75. Burges C. A tutorial on support vector machines for pattern recognition. Data mining and Knowledge Discovery 1998;2:1–47.
76. Schölkopf, B.; Smola, A. Learning with Kernels. MIT Press; Cambridge, Massachussetts: 2001.
77. Frishman D, Argos P. Knowledge-based protein secondary structure assignment. Proteins: Struct. Func. Genet 1995;23:566–579.
78. Kraulis PJ. Molscript: a program to produce both detailed and schematic plots of protein structures. J. Appl. Crystallo 1991;24:946–950.

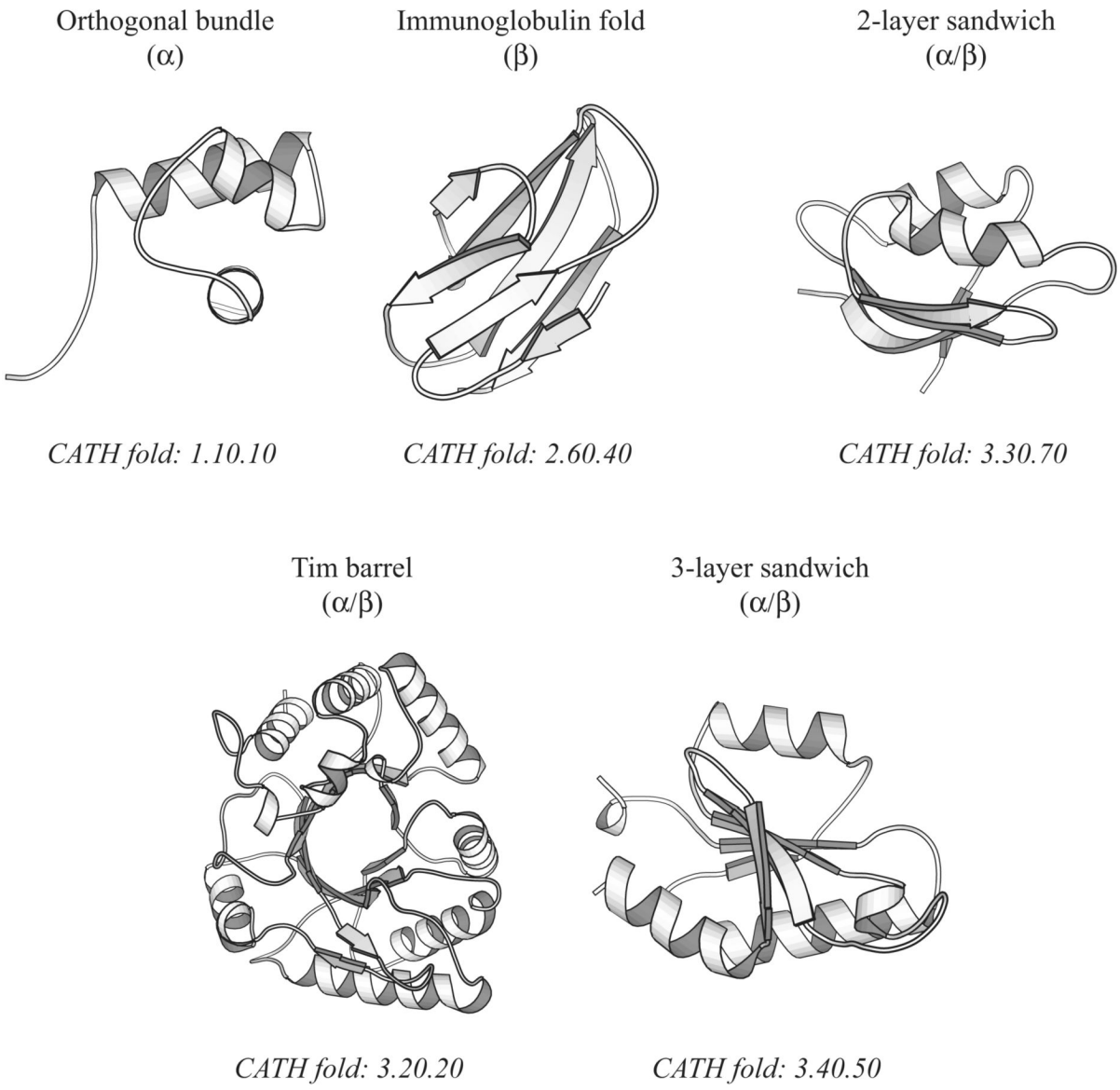

Fig. 1. Different 1D sequence representations of a toy protein structure

The native sequence (NS) is simply the primary sequence, or list of the amino acids forming the protein, from Nter to Cter. The SSE sequence (SSES) defines the secondary structure element for each residue in the protein, according to STRIDE.[77] Three types of SSE are considered: helices (H), strands or extended conformations (E) and coils (C). The Fragment sequence (FS) is the sequence of fragments found in the best-fit reconstruction model for the protein (shown here as a stick model, with different colors for each type of fragments)). The fragments are drawn from a library of 20 fragments of size 4 residues. They are uniquely identified with a letter from A to T. Two types of fragment sequences are considered: the local sequence (LFS), that ensures the best local fit of the fragment, and the global sequence (GFS), that ensures the best global fit of the structural model to the complete protein structure. Note that by construction, the fragment-based sequences are shorter than the two other sequences (the four first residues being defined by the first fragment). Models were drawn using Pymol (http://www.pymol.org).

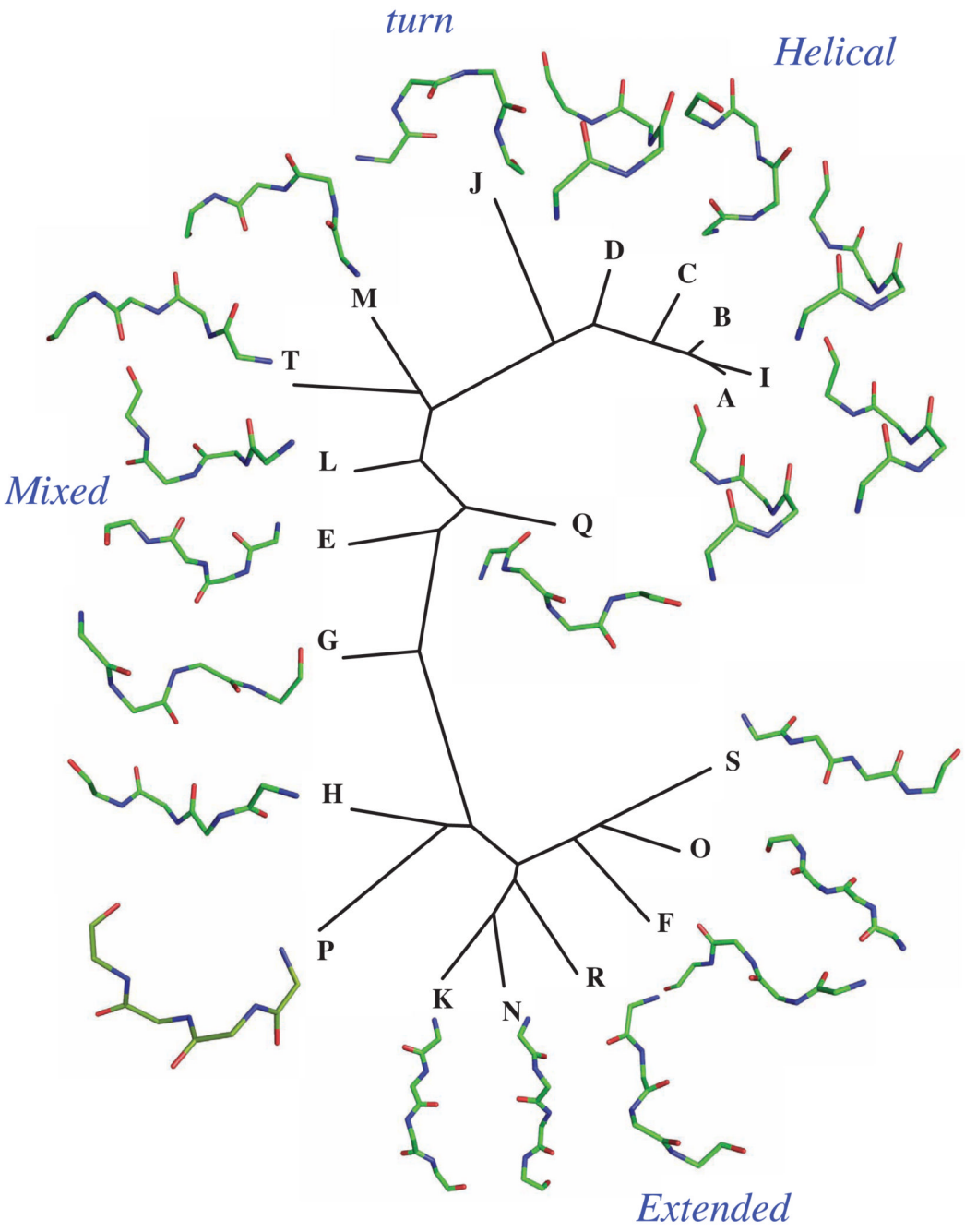

Fig. 2. Representatives of the five fold classes in our test set

The arc repressor mutant, subunit A fold (CATH 1.10.10) is a common orthogonal helix bundle, found for example in the DNA-binding domain of the human telomeric protein HTRF1 (CATH code 1ba500). The immunoglobulin-like fold is a β sandwich, found in many immunoglobulin-like proteins, such as the rat CD4 protein (CATH code 1cid02). The TIM barrel is a very common $\alpha - \beta$ fold, shown in narbonin, a plant seed protein (CATH code 1nar00). The $\alpha - \beta$ plait fold is a two layer sandwich, shown here in MERP, a mercury binding protein (CATH code 2hqi00). The Rossmann fold is a very common 3-layer sandwich fold in the mixed $\alpha - \beta$ class, found for example in the glycyl-tRNA synthetase from thermus thermophilus (CATH code 1atiB2). All images were generated using MOLSCRIPT.[78]

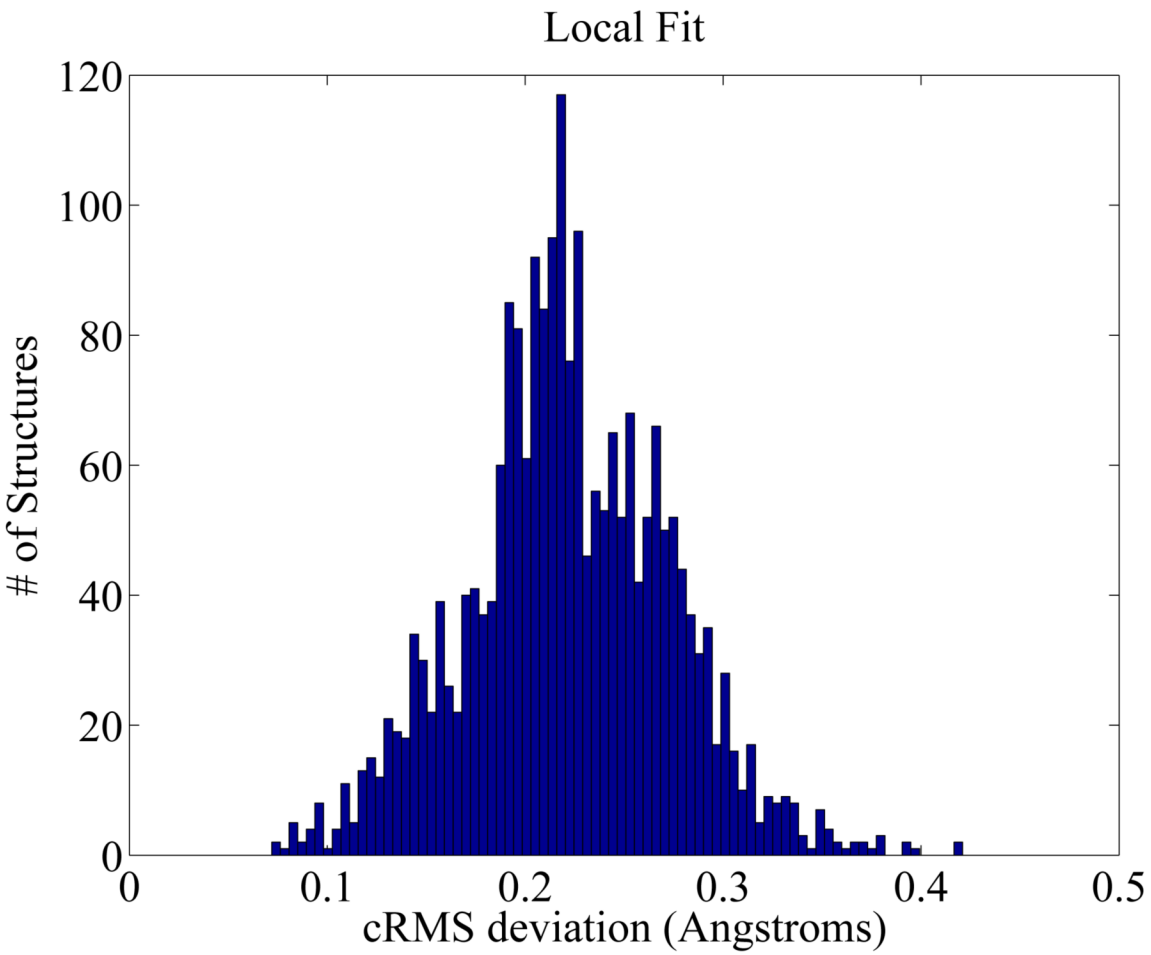

Fig.3. Our library of 20 fragments of size 4 residues

Images are generated using Pymol. The fragments have been organized around a tree based on similarity (cRMS), computed with the program Phylip.[49]

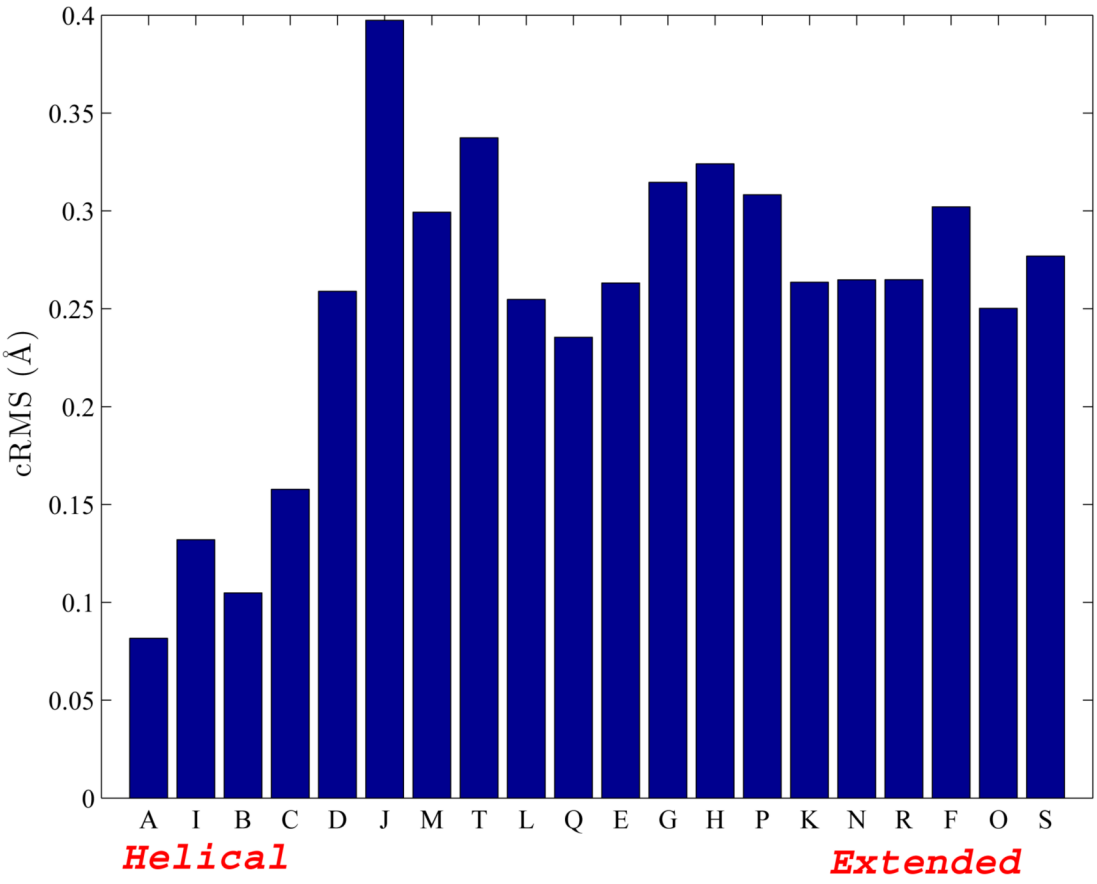

Fig. 4. Testing the accuracy of local structural sequences

Histogram of the mean cRMS between all 4 consecutive residue fragments of a protein and their best local-fit fragments in the library, for all proteins in CATH2225. The average cRMS over all fragments is 0.22.

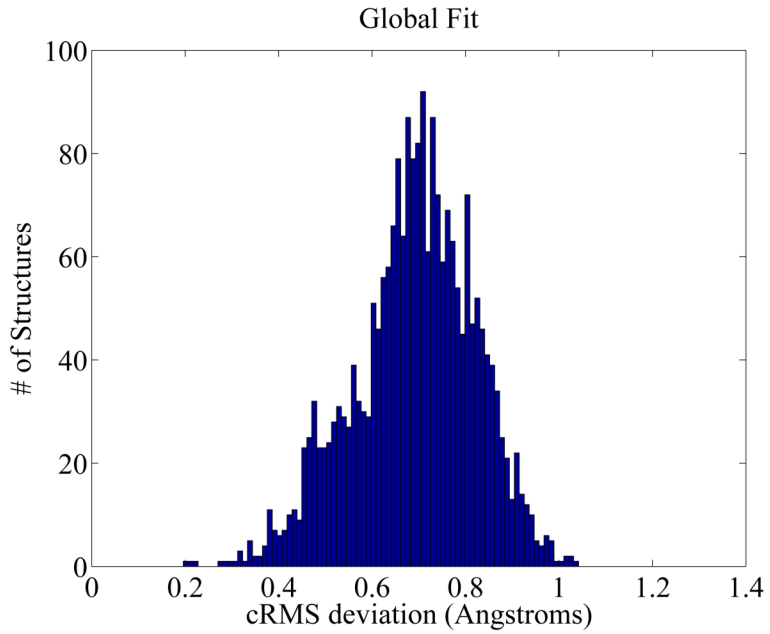

Fig. 5. Best local fits for all 20 structural fragments

Mean quality of the best local-fit for each of the 20 fragments of size 4. The fragments have been ordered based on similarity (see figure 3), with helical-like fragments on the left, and extended fragments on the right.

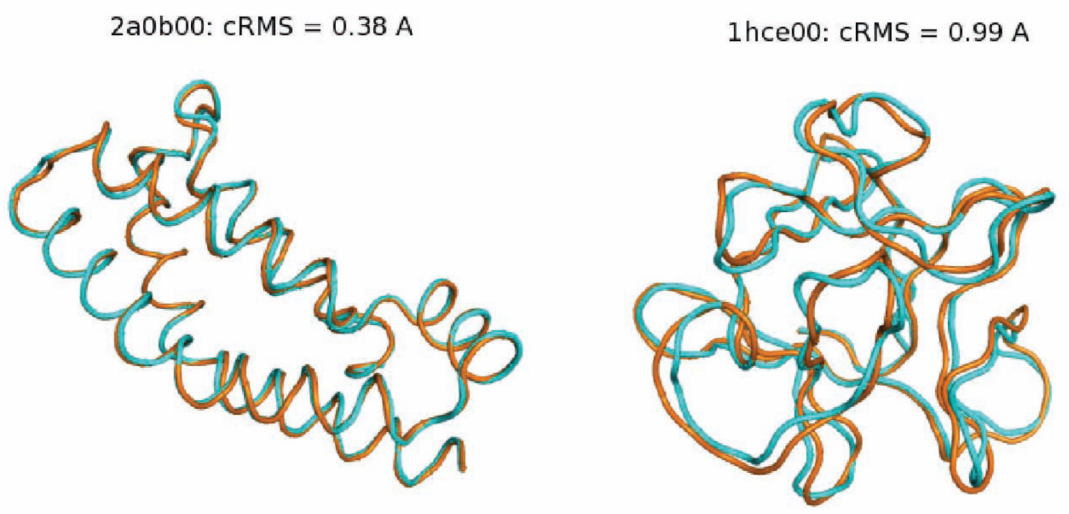

Fig. 6. Testing the accuracy of global structural sequences

Histogram of global-fit cRMS deviations between the model built from the global structural sequence and the native structure for all proteins in CATH2225. Structural sequences were built from a library of 20 fragments of size 4 residues, using a soft greedy approach with a heap size *Nkeep* = 4000 (see text for details). The average cRMS is 0.69.

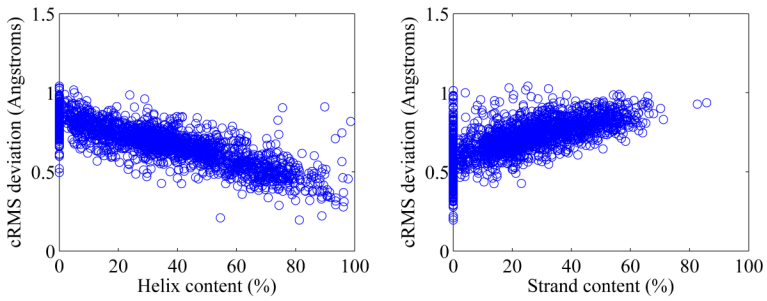

Fig. 7. Examples of protein structure reconstruction

The global-fit algorithm was used to reconstruct 2a0b00, a fully protein, and 1hce00, a fully β protein, using a library of 20 fragments of length 4 residues. The native structure is shown in orange, and the fragment-based structure in cyan.

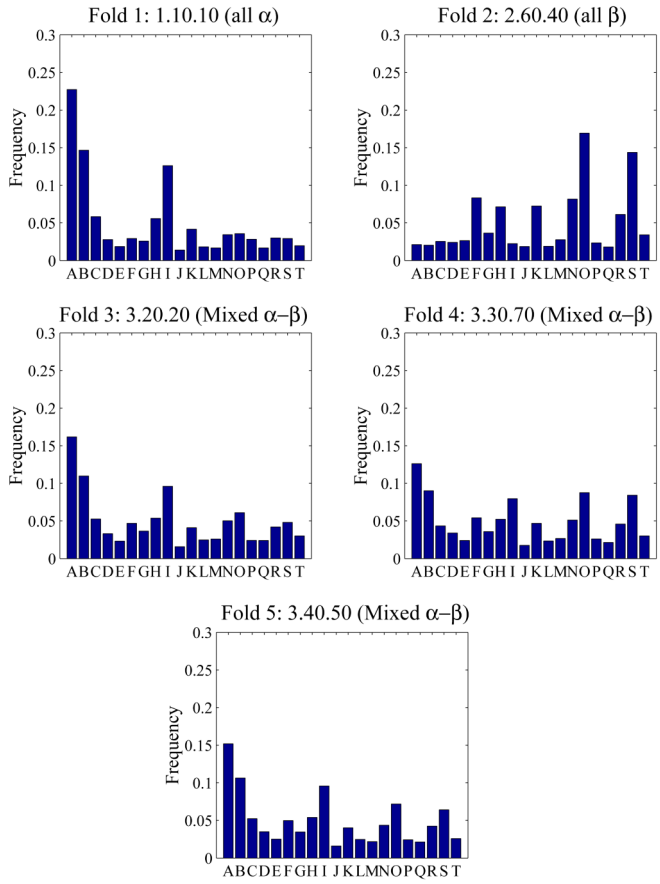

Fig. 8. Accuracy of structural sequences *versus* secondary structure content

The global-fit accuracy of the model reconstructed from the structural sequence of a protein is plotted against the α-helix content (left), and β-strand content (right) of the protein.

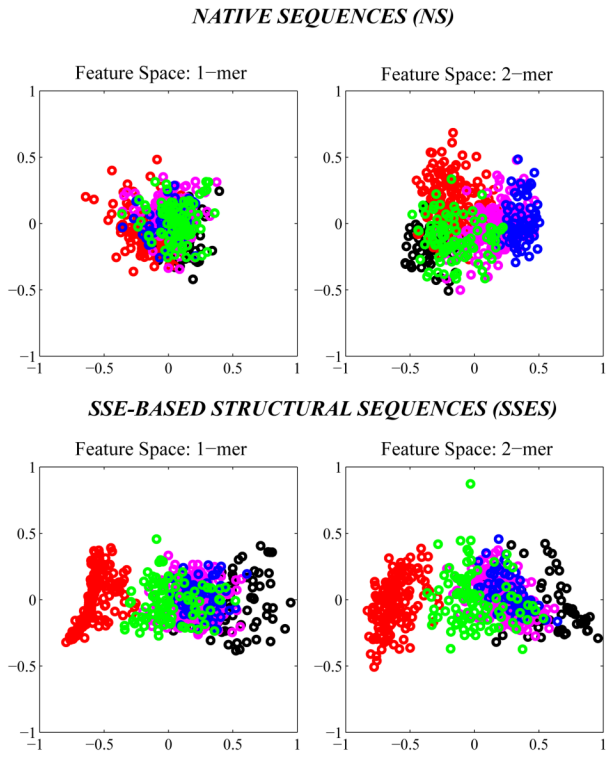

Fig. 9.
Distribution of fragment usage in the global structural sequences for five distinct folds.

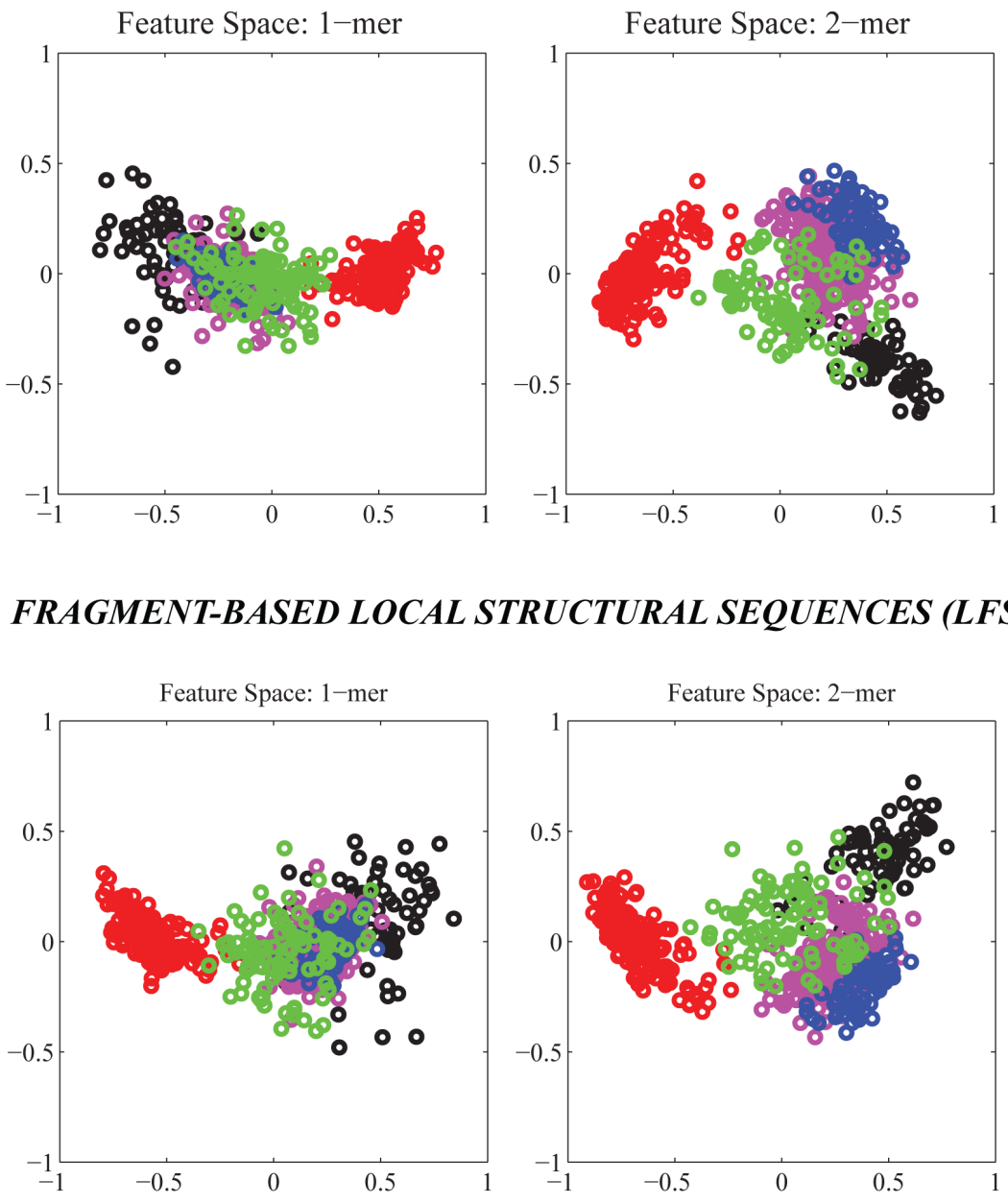

Fig. 10. 2D representations of the protein sequence feature spaces

Each protein in our dataset is assigned four different 1D sequences: its native sequence (NS), the sequence of its secondary structure elements (SSES), and the two structural sequences derived from a structural alphabet of fragments, global (GFS), and local (LFS). These sequences are mapped onto a feature space indexed on their 1-mer or 2-mer content. Each protein is assigned a color based on the fold it belongs to: black for 1.10.10 (all α proteins), red for 2.60.40 (all β proteins), and blue, magenta and green for 3.20.20, 3.30.70 and 3.40.50, respectively (mixed $\alpha - \beta$ folds). 2D projections of these spaces were generated using metric Multi-Dimensional Scaling.[51]

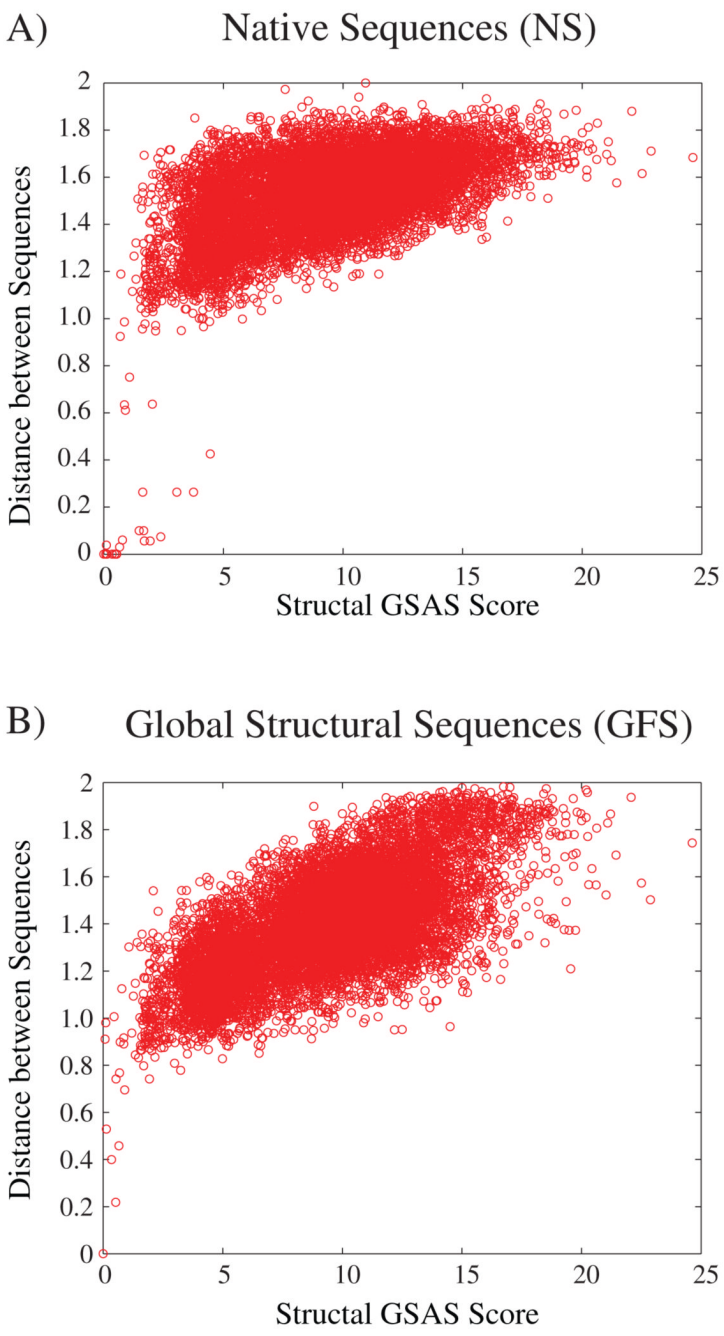

Fig. 11. The relationship between sequence similarity and structure similarity

We use 10,000 pairs of proteins in CATH2225. For each pair of proteins, the L1-distance between the projections of the native sequence (A) and global structural sequence GF (B) in the 2-mer feature space is plotted versus the GSAS score between their structures (GSAS is computed as 100 * *cRMS*/(*N* – *Ngap*), where *N* is the number of equivalent residues and *Ngap* the number of gaps in the alignment.[19] No relation is observed for the native sequence while a linear relationship is observed for GFS, with a correlation coefficient R=0.7.

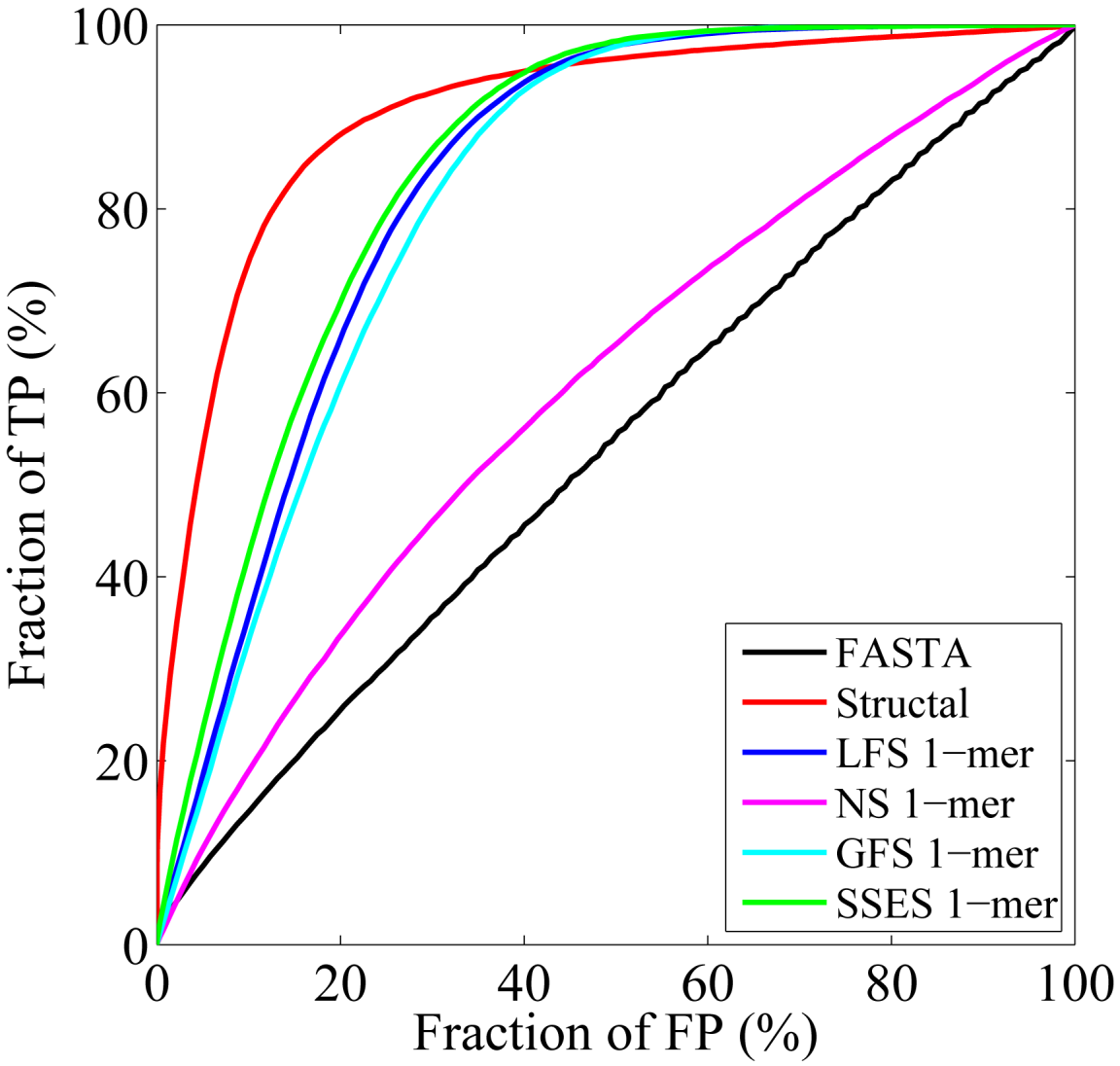

Fig. 12. ROC analysis comparing similarity measures in protein sequence feature spaces for remote homology detection

We compare the ability of STRUCTAL, a protein structure alignment program, FASTA, a sequence alignment program, and 4 classifiers based on the native sequence, the SSE sequence and the global and local structural sequences of a protein mapped in the 1-mer feature space to detect structural similarities in a set of 605 proteins. “True” relationships are defined by CATH topologies. Curves close to the first diagonal (such as the ROC curve for FASTA) indicates poor performance, while the upper most curves (such as STRUCTAL) reveal good performance.

Table 1

Effect of heap size on Global fit

The average quality of structure reconstruction (given as cRMS) over all proteins in CATH2225 is given for various heap sizes *Nkeep* used by the global fit algorithm. We also give the CPU times (in seconds on a Intel Core2 processor at 2.8 GHz) required for the reconstruction of one protein of 759 residues (1cm5B0) for the different values of *Nkeep*.

Nkeep	Average cRMS (Å)	CPU time (s)
10	0.82	0.45
100	0.72	3.9
1000	0.70	21.6
1000	0.70	41.7
4000	0.69	174.1

Table 2
ROC scores for homology detections based on protein feature spaces

The ROC score is the area below the ROC curves that plots the rate of TP versus the rate of FP for a given homology detection method. High scores are always better.

	Feature space: 1-mer		Feature space: 2-mer	
Sequence	**L1 distance**	**Kernel distance**	**L1 distance**	**Kernel distance**
Native Sequence	0.61	0.61	0.65	0.60
SSE Sequence	0.84	0.85	0.85	0.84
GLobal Structural Sequence	0.82	0.83	0.80	0.76
Local Structural Sequence	0.83	0.84	0.82	0.79

Table 3
Classifying protein structures based on a sequence-based feature space

Protein sequences are mapped in a high-dimensional space based on its 1-mer or 2-mer content. Four types of sequences are considered: the native sequence, the sequence of its secondary structure elements, i and the global and local fragment-based structural sequences. Proteins are classified based on their shortest distance to a known fold, where the distance is either the L1 norm, or a kernel based distance. The accuracy is computed as the ratio of proteins correctly classified over the total number of test proteins.

	Feature space: 1-mer		Feature space: 2-mer	
Sequence	**L1 distance**	**Kernel distance**	**L1 distance**	**Kernel distance**
Native Sequence	46.6 ± 2.4	49.8 ± 2.6	54.1 ± 2.7	62.9 ± 2.4
SSE Sequence	65.8 ± 1.8	65.4 ± 1.9	67.1 ± 1.8	66.2 ± 2.0
GLobal Structural Sequence	64.5 ± 1.9	66.2 ± 1.8	70.2 ± 2.1	72.5 ± 2.5
Local Structural Sequence	69.2 ± 1.9	69.2 ± 2.0	74.3 ± 2.1	70.4 ± 2.5

Table 4

SVM-based classification of protein structures using structural sequences

Sequence	Feature space: 1-mer	Feature space: 2-mer
Native Sequence	58.7 ± 1.8	61.4 ± 2.2
SSE Sequence	74.2 ± 1.2	74.6 ± 1.2
Global Structural Sequence	75.5 ± 1.3	80.6 ± 1.7
Local Structural Sequence	76.8 ± 1.6	81.3 ± 1.7