Technical Report

Department of Computer Science
and Engineering
University of Minnesota
4-192 EECS Building
200 Union Street SE
Minneapolis, MN 55455-0159 USA

TR 07-014

fRMSDAlign: Protein Sequence Alignment Using Predicted Local Structure Information

Huzefa Rangwala and George Karypis

May 31, 2007

fRMSDAlign: Protein Sequence Alignment Using Predicted Local Structure Information

Huzefa Rangwala and George Karypis

Computer Science & Engineering, University of Minnesota, Minneapolis, MN 55455
rangwala@cs.umn.edu, karypis@cs.umn.edu

Abstract

As the sequence identity between a pair of proteins decreases, alignment strategies that are based on sequence and/or sequence profiles become progressively less effective in identifying the correct structural correspondence between residue pairs. This significantly reduces the ability of comparative modeling-based approaches to build accurate structural models. Incorporating into the alignment process predicted information about the local structure of the protein holds the promise of significantly improving the alignment quality of distant proteins. This paper studies the impact on the alignment quality of a new class of predicted local structural features that measure how well fixed-length backbone fragments centered around each residue-pair align with each other. It presents a comprehensive experimental evaluation comparing these new features against existing state-of-the-art approaches utilizing profile-based and predicted secondary-structure information. It shows that for protein pairs with low sequence similarity (less than 12% sequence identity) the new structural features alone or in conjunction with profile-based information lead to alignments that are considerably better than those obtained by previous schemes.

Keywords: sequence alignment, machine learning, comparative modeling

1 Introduction

Over the years a wide range of comparative modeling-based methods [23, 25, 28] have been developed for predicting the structure of a protein (target) from its amino acid sequence. The central idea behind these techniques is to align the sequence of the target protein to one or more template proteins and then construct the target's structure from the structure of the template(s) using the alignment(s) as reference.

The overall performance of comparative modeling approaches [30, 31] depends on how well the alignment, constructed by considering sequence and sequence-derived information, agrees with the structure-based alignment between the target and the template proteins. This can be quite challenging, as two proteins can have high structural similarity even though there exists very little sequence identity between them. This led to the development of sophisticated profile-based methods and scoring functions [8, 1, 5, 32, 10, 19] that allowed high-quality alignments between protein pairs whose sequence identities are as low as 20%. However, these profile-based methods become less effective for protein pairs with lower similarities. As a result, researchers are increasingly relying on alignment scoring methods that also incorporate various predicted structural information such as secondary structure, backbone angles, and protein blocks [12, 20, 7, 13].

Recently we developed machine-learning methods [21] that can accurately estimate the root mean squared deviation (RMSD) value of a pair of equal-length protein fragments (i.e., contiguous backbone segments) by considering only sequence and sequence-derived information. Our interest in solving this problem is motivated by the operational characteristics of various dynamic-programming-based [18, 29] protein structure alignment methods like CE [27] and MUSTANG [14] that score the aligned residues by computing the RMSD value of the optimal superimposition of the two fixed-length fragments centered around each residue. Thus, by being able to accurately predict the RMSD values of all these fragment-pairs from the protein sequence alone, we can enable the target-template alignment algorithms to use the same information as that used by the structure alignment methods.

In this paper we focus on studying the extent to which the predicted fragment-level RMSD (fRMSD) values can actually lead to alignment improvements. Specifically, we study and evaluate various alignment scoring schemes that use information derived from sequence profiles, predicted secondary structure, predicted fRMSD values, and their combinations. Results on two different datasets show that scoring schemes using the predicted fRMSD values alone and/or in combination with scores derived from sequence profiles lead to better alignments than those obtained by current state-of-the-art schemes that utilize sequence profiles and predicted secondary structure information, especially for sequence pairs having less than 12% sequence identity. In addition, we present two methods based on seeded alignments and iterative sampling that significantly reduce the number of fRMSD values that need to be predicted, without a significant loss in the overall alignment accuracy. This significantly reduces the computational requirements of the proposed alignment strategies.

The rest of the paper is organized as follows. Section 2 provides key definitions and notations used throughout the paper. Section 3 describes the datasets and the various computational tools used in this paper. Section 4 describes the scoring schemes used in our study and the various optimizations that we developed. Section 5 presents a comprehensive experimental evaluation of the methods developed. Finally, Section 6 summarizes the work and provides some concluding remarks.

2 Definitions and Notations

Throughout the paper we will use X and Y to denote proteins, x_i to denote the ith residue of X, and $\pi(x_i, y_j)$ to denote the *residue-pair* formed by residues x_i and y_j.

Given a protein X of length n and a user-specified parameter v, we define $v\texttt{frag}(x_i)$ to be the $(2v+1)$-length contiguous substructure of X centered at position i ($v < i \leq n - v$). These substructures are commonly referred to as fragments [27, 14]. Given a residue-pair $\pi(x_i, y_j)$, we define $f\texttt{RMSD}(x_i, y_j)$ to be the *structural similarity score* between $v\texttt{frag}(x_i)$ and $v\texttt{frag}(y_j)$. This score is computed as the root mean square deviation between the pair of substructures after optimal superimposition. Finally, we define the *fRMSD estimation* problem as that of estimating the $f\texttt{RMSD}(x_i, y_j)$ score for a given residue-pair $\pi(x_i, y_j)$ by considering only information derived from the amino acid sequence of X and Y.

3 Materials

3.1 Datasets

We evaluate the accuracy of the alignment schemes on two datasets. The first dataset, referred to as the ce_ref dataset, was used in a previous study to assess the performance of different profile-profile scoring functions for aligning protein sequences [5]. The ce_ref dataset consists of 581 alignment pairs having high structural similarity but low sequence identity ($\leq 30\%$). The gold standard reference alignment was curated from a consensus of two structure alignment programs: FSSP [11] and CE [27]. The second dataset, referred to as the mus_ref dataset, was derived from the SCOP 1.57 database [17]. This dataset consists of 190 protein pairs with an average sequence identity of 9.6%. Mustang [14] was used to generate the gold standard reference alignments.

To better analyze the performance of the different alignment methods, we segmented each dataset based on the pairwise sequence identities of the proteins that they contain. We segmented the ce_ref dataset into four groups, of sequence identities in the range of 6-12%, 12-18%, 18-24%, and 24-30% that contained 15, 140, 203, and 223 pairs of sequences, respectively. We segmented the mus_ref dataset into three groups, of sequence identities in the range of 0-6%, 6-12%, and 12-30% that contained 76, 67, and 47 pairs of sequences, respectively. Note that the three groups of the mus_ref are highly correlated with the bottom three levels of the SCOP hierarchy, with most pairs in the first group belonging to the same fold but different superfamily, most pairs in the second group belonging to the same superfamily but different family, and most pairs in the third group belonging to the same family.

3.2 Evaluation Methodology

We evaluate the quality of the various alignment schemes by comparing the differences between the generated candidate alignment and the reference alignment generated from structural alignment programs [5, 24, 6]. As a measure of alignment quality, we use the Cline Shift score (CS) [2] to compare the reference alignments with the candidate alignments. The CS score is designed to penalize both under- and over-alignment and crediting the parts of the generated alignment that may be shifted by a few positions relative to the reference alignment [5, 2, 22]. The CS score ranges from a small negative value to 1.0, and is symmetric in nature. We also assessed the performance on the standard Modeler's (precision) and Developer's (recall) score [24], but found similar trends to the CS score and hence do not report the results here.

3.3 Profile Generation

The profile [1] of a sequence X of length n is represented by two $n \times 20$ matrices, namely the position-specific scoring matrix $\mathcal{P}_X$ and the position-specific *frequency* matrix $\mathcal{F}_X$. These profiles capture evolutionary information for a sequence. The $\mathcal{F}_X(i)$ and $\mathcal{P}_X(i)$ are the ith column of X's position-specific scoring and frequency matrices. For our study, the profile matrices $\mathcal{P}$ and $\mathcal{F}$ were generated using PSI-BLAST [1] with the following parameters: `blastpgp -j 5 -e 0.01 -h 0.01`. The PSI-BLAST was performed against NCBI's nr database that was downloaded in November of 2004 and contained 2,171,938 sequences.

3.4 Secondary Structure Prediction

For a sequence X of length n we predict the secondary structure and generate a position-specific secondary structure matrix $\mathcal{S}_X$ of length $n \times 3$. The (i, j) entry of this matrix represents the strength of the amino acid residue at position i to be in state j, where $j \in (0, 1, 2)$ corresponds to the three secondary structure elements: alpha helices (H), beta strands (E), and coil regions (C). We use the state-of-the-art secondary structure prediction server YASSPP [13] (default parameters) to generate the $\mathcal{S}$ matrix. The values of the $\mathcal{S}$ matrix are the output of the three one-versus-rest SVM classifiers trained for each of the secondary structure elements.

3.5 fRMSD Estimation

To estimate the $f\texttt{RMSD}$ scores for a residue-pair $\pi(x_i, y_j)$ we used the recently developed fRMSDPredprogram [21]. The fRMSDPredprogram uses an ϵ-SVR learning methodology to estimates the $f\texttt{RMSD}$ score of a reside-pair $\pi(x_i, y_j)$ by taking into account the profile and the predicted secondary structure of a fixed-length window around the x_i and y_j residues. The ϵ-SVR estimation technique deploys a novel second-order pairwise exponential kernel function which shows superior results in comparison to the radial basis kernel function.

The ϵ-SVR implementation used the publicly available support vector machine tool SVM^{light} [26] which has an efficient ϵ-SVR implementation. We used the defaults for regularization and regression tube width parameters. The fRMSDPredprogram was trained on a dataset consisting of 1117 protein pairs derived from the SCOP 1.57 database. This training dataset was used in previous studies [21, 19], and no two protein domains in the dataset shared greater than 75% sequence identity. For each protein pair in the train dataset we use the standard Smith-Waterman [29] algorithm to generate the residue-pairs for which we compute the $f\texttt{RMSD}$ score by considering fragment lengths of seven.

3.6 Gap Modeling and Shift Parameters

For all the different scoring schemes, we use a local alignment framework with an affine gap model, and a zero-shift parameter [32] to maintain the necessary requirements for a good optimal alignment [9]. We optimize the gap modeling parameters (gap opening (*go*), gap extension (*ge*)), the zero shift value (*zs*), and weights on the individual scoring matrices for integrating them to obtain the highest quality alignments for each of the schemes. Having optimized the alignment parameters on the ce_ref dataset, we keep the alignment parameters unchanged for evaluation on the mus_ref dataset.

4 Methods

4.1 Scoring Schemes

We use the standard Smith-Waterman based local alignment [29] algorithm in our methods. The different alignment schemes vary in the computation of the position-to-position similarity scores between residue-pairs.

4.1.1 Profile-Profile Scoring Scheme Many different profile-profile scoring functions [16, 32, 15] have been developed for determining the similarity between a pair of profile columns (i.e., residue-pairs). We use one of the best performing profile-profile scoring functions called PICASSO [10, 16], which computes the similarity between the ith position of protein's X profile and the jth position of the protein's Y profile as $\langle \mathcal{F}_X(i), \mathcal{P}_Y(j) \rangle + \langle \mathcal{F}_Y(j), \mathcal{P}_X(i) \rangle$. The operator $\langle , \rangle$ denotes a dot-product operation. We will refer to this scoring scheme as *prof*.

4.1.2 Predicted Secondary Structure-based Scoring Scheme For a given residue-pair $\pi(x_i, y_j)$ we compute the similarity score based on the predicted secondary structure information as a dot-product of the ith row of $\mathcal{S}_X$ and the jth row of $\mathcal{S}_Y$, i.e., $\langle \mathcal{S}_X(i), \mathcal{S}_Y(j) \rangle$. This approach of incorporating secondary structure information along with profiles, has been shown to significantly improve the alignment quality [20]. We will refer to this scoring scheme as *ss*.

4.1.3 fRMSD-based Scoring Scheme For a given residue-pair $\pi(x_i, y_j)$, we use the fRMSDPredprogram [21] to estimate its $f\texttt{RMSD}(x_i, y_j)$ score. Since this score is actually a distance, we convert it into a similarity score using the transformation: $\log(\alpha / f\texttt{RMSD}(x_i, y_j))$. This transformation assigns positive values to residue-pairs $\pi(x_i, y_j)$ having an estimated $f\texttt{RMSD}$ score that is less than α. For the purposes of this study the α parameter was set to one, because we observed that the residue-pairs $\pi(x_i, y_j)$ with $f\texttt{RMSD}(x_i, y_j)$ score of less than one are more likely to be structurally aligned. We will refer to this scoring scheme as *frmsd*.

4.2 Combination Schemes

Besides the above scoring schemes, we also investigated their combinations. We used a weighted combination of the profile-based, predicted secondary, and $f\texttt{RMSD}$-based scoring schemes to compute a similarity score for a residue pair $\pi(x_i, y_j)$. In this approach the similarity score for a residue-pair $\pi(x_i, y_j)$, using the *prof* and *frmsd* scoring schemes is given by

$$\frac{w * \textit{prof}(i,j)}{maxP} + \frac{(1-w) * \textit{frmsd}(i,j)}{maxF}, \qquad (1)$$

where *prof*(i,j) and *frmsd*(i,j) represent the PICASSO and $f\texttt{RMSD}$ scores for $\pi(x_i, y_j)$, respectively. The value $maxP$ ($maxF$) is the maximum absolute value of all *prof*-based (*frmsd*-based) residue-pair scores between the sequences and is used to normalize the different scores prior to addition. The parameter w defines the weighting for different parts of the scoring function after normalization. The optimal weight parameter w, was determined by varying w from 0.0 to 1.0 with increments of 0.1. This parameter was optimized for the ce_ref dataset, and the same value was then used for the mus_ref dataset.

A similar approach is used to combine *prof* with *ss* and *frmsd* with *ss*. In case of the *frmsd +prof +ss* there are two weight parameters that need to be optimized.

We will denote the various combination schemes by just adding their individual components, e.g., *frmsd +prof* will refer to the scheme that uses the scores obtained by *frmsd* and *prof*.

4.3 Speedup Optimization

For a residue-pair, we can compute the PICASSO- and secondary structure-based scores using two and one dot-product operations, respectively. In comparison, the $f\texttt{RMSD}$ score needs $|SV|$ dot-product operations, where $|SV|$ is the number of support vectors determined by the ϵ-SVR optimization method. Hence, the *frmsd* alignment scheme has a complexity of at least $O(|SV|)$, which is significantly higher than that of the *prof* and *ss* alignment schemes. To reduce these computational requirements we developed two heuristic alignment methods that require the estimation of only a fraction of the total number of residue pairs.

4.3.1 Seeded Alignment The first method combines the banded alignment approach and the seed alignment technique [9] and is performed in three steps. In the first step, we generate an initial alignment, referred to as the *seed alignment*, using the Smith-Waterman algorithm and the *prof +ss* scoring scheme. In the second step, we estimate the $f\texttt{RMSD}$ scores for all residue-pairs within a fixed number of residues from the seed alignment, i.e., a *band* around the seed alignment. Finally, in the third step, we compute the optimal local alignment in the restricted band around the initial seed alignment. The computed *frmsd* alignment lies within a fixed band around the *prof +ss* alignment and will be effective if the original *frmsd* alignment and the *prof +ss* alignments are not very far away from each other. The complexity of this method can be controled by selecting bands of different sizes. We refer to this method as the seeded alignment technique. Note that this method is essentially a refinement technique on the initial

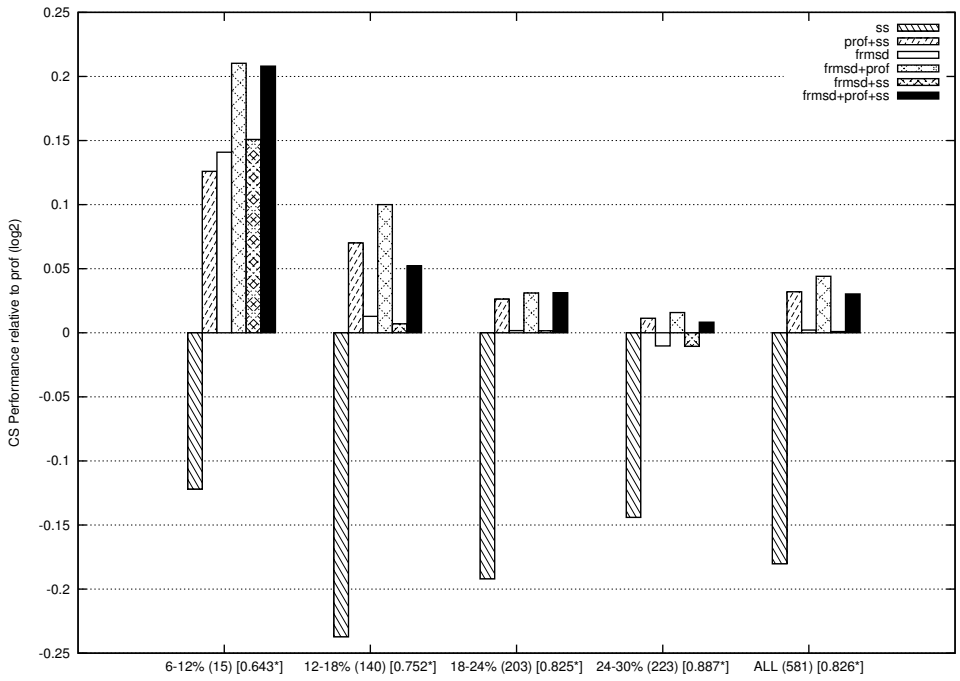

Figure 1: Relative CS Scores on the ce_ref dataset. For each segment we display the range of percent sequence identity, the number of pairs in the segment, and the average CS score of the baseline prof scheme.

seed alignment.

4.3.2 Iterative Sampling Alignment The second method employs an iterative sampling procedure to optimize the speed of the *frmsd* alignment. The basic idea is fairly similar to the seeded alignment. At iteration i, we estimate 1 out of R_i $f\texttt{RMSD}$ scores in the dynamic-programming matrix for those residue-pairs that lie within the banded region of size K_i around the seed alignment generated in step $i-1$. K_i and R_i denote the band size and the sampling rate at iteration i, respectively. Using the estimated $f\texttt{RMSD}$ scores, an alignment is produced at step i which serves as the seed alignment for step $i+1$. The band size is reduced by half, whereas the sampling rate is doubled at each step (i.e., R_i will be halved), effectively increasing the number of points in the dynamic-programming matrix to be estimated within a confined band. The first iteration can be assumed to have the initial seed as the main diagonal with a band size covering the entire dynamic-programming matrix.

5 Results

We performed a comprehensive study to evaluate the accuracy of the alignments obtained by the scoring scheme derived from the estimated *frmsd* values against those obtained by the *prof* and *ss* scoring schemes and their combinations. These results are summarized in Figures 1 and 2, which show the accuracy performance of the different scoring schemes on the ce_ref and mus_ref datasets, respectively. The alignment accuracy is assessed using the average CS scores across the entire dataset and at the different pairwise sequence identity segments. To better illustrate the differences between the schemes, the results are presented relative to the CS score obtained by the *prof* alignment and are shown on a $\log_2$ scale.

Analyzing the performance of the different scoring schemes we see that most of those that utilize predicted information about the protein structure (*ss*, *frmsd*, and combinations involving them and *prof*) lead to substantial improvements over the *prof* scoring scheme for the low sequence identity segments. However, the relative advantage of these schemes somewhat diminishes for the segments that have higher pairwise sequence identities. In fact, in the case of the 12%–30% segment for mus_ref, most of these schemes perform worse than *prof*. This result is not surprising, and confirms our earlier discussion in Section 1.

Comparing the *ss* and *frmsd* scoring schemes, we see that the latter achieves consistently and substantially better performance across the two datasets and sequence identity segments. For instance, for the first segment of ce_ref (sequence identities in the range of 6%–12%), *frmsd*'s CS score is 20% higher than that achieved by the *ss* scoring scheme. In the first segment of mus_ref dataset (sequence identity in the range of 0%–6%), *frmsd*'s CS score is 33% higher than achieved by the *ss* scoring scheme, and is 19% higher for the second segment (sequence identity in the range of 6%–12%).

Comparing most of the schemes based on *frmsd* and its combinations with the other scoring schemes we see that for the segments with low sequence identities they achieve the

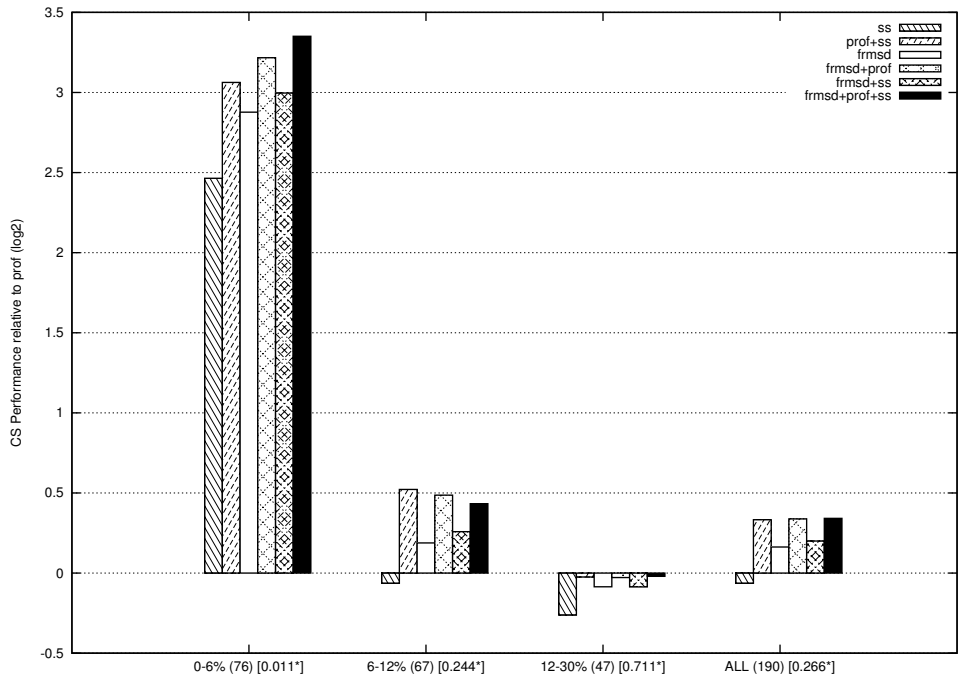

Figure 2: Relative CS Scores on the mus_ref dataset. For each segment we display the range of percent sequence identity, the number of pairs in the segment, and the average CS score of the baseline prof scheme.

best results. Among them, the *frmsd +prof* scheme achieves the best results for ce_ref, whereas the *frmsd +prof +ss* performs the best for mus_ref. For the first segments of ce_ref and mus_ref, both of these schemes perform 6.1% and 27.8% better than *prof +ss*, respectively, which is the best non-*frmsd*-based scheme. Moreover, for many of these segments, the performance achieved by *frmsd* alone is comparable to that achieved by the *prof +ss* scheme. Also, comparing the results obtained by *frmsd* and *frmsd +ss* we see that by adding information about the predicted secondary structure the performance does improve. In the case of the segments with somewhat higher sequence identities, the relative advantage of *frmsd +prof* diminishes and becomes comparable to *prof +ss*.

Finally, comparing the overall performance of the various schemes on the ce_ref and mus_ref datasets, we see that *frmsd +prof* is the overall winner as it performs the best for ce_ref and similar to the best for mus_ref.

5.1 Comparison to Other Alignment Schemes

Since the ce_ref dataset has been previously used to evaluate the performance of various scoring schemes we can directly compare the results obtained here with those presented in [5]. In particular, according to that study, the best PSI-BLAST-profile based scheme achieved a CS score of 0.805, which is considerably lower than the CS scores of 0.854 and 0.845 obtained by the *frmsd +prof* and *prof +ss*, respectively.

Also, to ensure that the CS scores achieved by our schemes on the mus_ref dataset are reasonable, we compared them against the CS scores obtained by the state-of-the-art CONTRALIGN [3] and ProbCons [4] schemes. These schemes were run locally using the default parameters. CONTRALIGN and ProbCons achieved average CS scores of 0.197 and 0.174 across the 190 alignments, respectively. In comparison the *frmsd* scheme achieved an average CS score of 0.299, whereas *frmsd +prof* achieved an average CS score of 0.337.

5.2 Optimization Performance

We also performed a sequence of experiments to evaluate the extent to which the two runtime optimization methods discussed in Section 4.3 can reduce the number of positions whose $f\texttt{RMSD}$ needs to be estimated while still leading to high-quality alignments. These results are shown in Fig. 3, which shows the CS scores obtained by the *frmsd* scoring scheme on the ce_ref dataset as a function of the percentage of the residue-pairs whose $f\texttt{RMSD}$ scores were actually estimated. Also, the figure shows the average CS score achieved by the original (not sampled) *frmsd* scheme.

These results show that both the seeded and iterative sampling procedures generate alignments close to the alignment generated from the original complete scheme. The average CS scores of the seeded and iterative sampling alignment by computing just 6% of the original *frmsd* matrix is 0.822 and 0.715, respectively. The average CS score of the original *frmsd* scheme is 0.828. Hence, we get competitive scores by

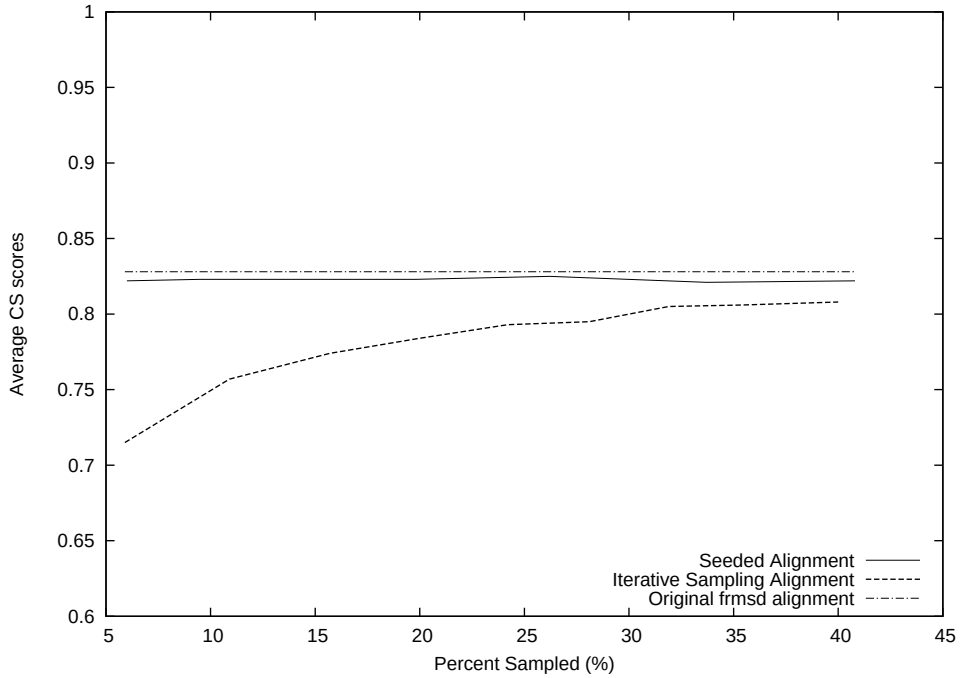

Figure 3: Speedup using the Seeding and Sampling Alignment Procedure on the ce_ref dataset.

our sampling procedures for almost a 20 fold speedup. The seeded based technique shows better performance compared to the iterative sampling technique.

6 Conclusion

In this paper we evaluated the effectiveness of using estimated fRMSD scores to aid in the alignment of protein sequences. Our results showed that the structural information encoded in these estimated scores are substantially better than the corresponding information in predicted secondary structures and when coupled with existing state-of-the-art profile scoring schemes, they lead to considerable improvements in aligning protein pairs with very low sequence identities.

This approach of estimating the fragment-level RMSD is of similar spirit to learning a profile-profile scoring function to differentiate related and unrelated residue pairs using artificial neural networks [19]. The results reported [19] found that the use of resulting profile-profile scoring functions did not assist in fold recognition.

7 Acknowledgment

This work was supported by NSF EIA-9986042, ACI-0133464, IIS-0431135, NIH RLM008713A, the Digital Technology Center and the Minnesota Supercomputing Institute at the University of Minnesota. A great thanks to Chris Kauffman and Kevin DeRonne for helping me revise this manuscript.

References

[1] S. F. Altschul, L. T. Madden, A. A. Schffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Research*, 25(17):3389–402, 1997.

[2] M. Cline, R. Hughey, and K. Karplus. Predicting reliable regions in protein sequence alignments. *Bioinformatics*, 18:306–314, 2002.

[3] C. B. Do, S. S. Gross, and S. Batzoglou. Contralign: Discriminative training for protein sequence alignment. *In Proceedings of the Tenth Annual International Conference on Computational Molecular Biology (RECOMB)*, 2006.

[4] C. B. Do, M. S. P. Mahabashyam, and S. Batzoglou. Probcons: Probabilistic consistency-based multiple sequence alignment. *Genome Research*, 15:330–340, 2005.

[5] R. Edgar and K. Sjolander. A comparison of scoring functions for protein sequence profile alignment. *BIOINFORMATICS*, 20(8):1301–1308, 2004.

[6] A. Elofsson. A study on protein sequence alignment quality. *PROTEINS:Structure, Function and Genetics*, 46:330–339, 2002.

[7] C. Etchebest, C. Benros, S. Hazout, and A. G. deBrevern. A structural alphabet for local protein structures: improved prediction methods. *Proteins: Struc-*

ture, Function, and Bioinformatics, 59(4):810–827, 2005.

[8] M. Gribskov and N. Robinson. Use of receiver operating characteristic (roc) analysis to evaluate sequence matching. *Computational Chemistry*, 20:25–33, 1996.

[9] Dan Gusfield. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*. Cambridge University Press, New York, NY, 1997.

[10] A. Heger and L. Holm. Picasso:generating a covering set of protein family profiles. *Bioinformatics*, 17(3):272–279, 2001.

[11] L. Holm and C. Sander. Mapping the protein universe. *Science*, 273(5275):595–602, 1996.

[12] D. T. Jones, W. R. Taylor, and J. M. Thorton. A new approach to protein fold recognition. *Nature*, 358:86–89, 1992.

[13] George Karypis. Yasspp: Better kernels and coding schemes lead to improvements in svm-based secondary structure prediction. *Proteins: Structure, Function and Bioinformatics*, 64(3):575–586, 2006.

[14] A. S. Konagurthu, J. C. Whisstock, P. J. Stuckey, and A. M. Lesk. Mustang: a multiple structural alignment algorithm. *Proteins: Structure, Function, and Bioinformatics*, 64(3):559–574, 2006.

[15] M. Marti-Renom, M. Madhusudhan, and A. Sali. Alignment of protein sequences by their profiles. *Protein Science*, 13:1071–1087, 2004.

[16] D. Mittelman, R. Sadreyev, and N. Grishin. Probabilistic scoring measures for profile-profile comparison yield more accurate short seed alignments. *Bioinformatics*, 19(12):1531–1539, 2003.

[17] A. G. Murzin, S. E. Brenner, T. Hubbard, and C. Chothia. Scop: a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology*, 247:536–540, 1995.

[18] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48:443–453, 1970.

[19] T. Ohlson and A. Elofsson. Profnet, a method to derive profile-profile alignment scoring functions that improves the alignments of distantly related proteins. *BMC Bioinformatics*, 6(253), 2005.

[20] J. Qiu and R. Elber. Ssaln: An alignment algorithm using structure-dependent substitution matrices and gap penalties learned from structurally aligned protein pairs. *Proteins: Structure, Function, and Bioinformatics*, 62(4):881–891, 2006.

[21] H. Rangwala and G. Karypis. frmsdpred: Predicting local rmsd between structural fragments using sequence information. In *To Appear in Proceedings of the 2007 International Conference on Computational Systems Bioinformatics*, 2007.

[22] H. Rangwala and G. Karypis. Incremental window-based protein sequence alignment algorithms. *Bioinformatics*, 23(2):e17–23, 2007.

[23] R. Sanchez and A. Sali. Advances in comparative protein-structure modelling. *Current Opinion in Structural Biology*, 7(2):206–214, 1997.

[24] J. M. Sauder, J. W. Arthur, and R. L. Dunbrack. Large-scale comparison of protein sequence alignments with structural alignments. *Proteins*, 40:6–22, 2000.

[25] T. Schwede, J. Kopp, N. Guex, and M. C. Peltsch. Swiss-model: An automated protein homology-modeling server. *Nucleic Acids Research*, 31(13):3381–3385, 2003.

[26] B. Schlkopf, C. Burges, and A. Smola, editors. *Making large-Scale SVM Learning Practical*. Advances in Kernel Methods - Support Vector Learning. MIT Press, 1999.

[27] I. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (ce) of the optimal path. *Protein Engineering*, 11:739–747, 1998.

[28] J. Skolnick and D. Kihara. Defrosting the frozen approximation: Prospector–a new approach to threading. *Proteins: Structure, Function and Genetics*, 42(3):319–331, 2001.

[29] T. F. Smith and M. S. Waterman. Identification of common molecular subsequences. *Journal of Molecular Biology*, 147:195–197, 1981.

[30] C. Venclovas. Comparative modeling in casp5: Progress is evident, but alignment errors remain a significant hindrance. *Proteins: Structure, Function, and Genetics*, 53:380–388, 2003.

[31] C. Venclovas and M. Margelevicius. Comparative modeling in casp6 using consensus approach to template selection, sequence-structure alignment, and structure assessment. *Proteins: Structure, Function, and Bioinformatics*, 7:99–105, 2005.

[32] G. Wang and R. L. Dunbrack JR. Scoring profile-to-profile sequence alignments. *Protein Science*, 13:1612–1626, 2004.