Distance-based Indexing of Residue Contacts for Protein Structure Retrieval and Alignment

Ahmet Sacan[1,2], I. Hakki Toroslu[1], and Hakan Ferhatosmanoglu[2]
[1]Computer Engineering Dept.
Middle East Technical University
Ankara, Turkey
ahmet,toroslu@ceng.metu.edu.tr

[2]Dept. of Computer Sci. & Eng.
The Ohio State University
Columbus, OH, USA
hakan@cse.ohio-state.edu

Abstract— **New protein structures are continuously being determined with the hope of deriving insights into the function and mechanisms of proteins, and consequently, protein structure repositories are growing by leaps and bounds. However, we are still far from having the right methods for sensitive and effective use of the available structural data. The fact that current structural analysis tools are impractical for large-scale applications have given rise to several approaches that try to quickly identify candidate proteins worthy of further analysis. Nonetheless, these approaches do not provide the desired sensitivity of identifying important structural similarities.**

In this study, we propose a new protein structure retrieval method (RCIndex: Residue-Contacts Index) that is based on accurate and efficient identification of similar residue contacts from a database of available protein structures. By defining a metric distance function for biologically meaningful comparison of residue contacts, distance-based indexing is made applicable for quick retrieval of similar residue contact seeds. These seeds are extended into high scoring segment pairs, which induce structural superpositions. The results show that RCIndex is effective in not only identifying related proteins, but also producing remarkably high quality structural alignments that are comparable to or better than those produced by popular pairwise alignment tools. To the best of our knowledge, this is the first time the protein structure retrieval and alignment tasks are successfully handled together.

Availability: **A preliminary version of RCIndex is available as a web service at `http://bio.cse.ohio-state.edu/RCIndex`**

Keywords: **Protein structure, similarity search, alignment, distance-based indexing.**

I. Introduction

Over the past few decades, the philosophy and methodology of research in biological sciences have shifted remarkably to make use of *in-silico* modeling and analysis, besides the traditional *in-vivo* and *in-vitro* experimentation. This shift was primarily due to the increasing availability of bio-molecular sequence and structure data, as a result of the advent of high-throughput sequencing and structure determination techniques.

While there are widely accepted and utilized sequence similarity search tools, such as BLAST and PSI-BLAST [2], the same is not so true for the structural similarity search. Consequently, most of the research groups rely on purely sequence based analysis. On the other hand, it has been repeatedly declared that protein structural data can provide more detailed and informative answers for the function, biochemical mechanism, and evolutionary history of the proteins. Evidently, there is a need for more effective and widely available protein structure search and analysis tools, which forms the main motivation of this study.

The initial research on the analysis of protein structures focused mainly on comparison of two proteins by pairwise structure alignment. The pairwise alignment tries to find a solution to two inter-related problems: finding residue-residue (or atom-atom) correspondences between two protein structures, and finding the optimal translation-rotation matrix to superimpose these structures, where the optimality is measured by an error function (usually the root mean square deviation, RMSD, is used). While finding the optimal superposition for a *given* set of correspondences can be solved in linear time in the length of the proteins [15], solving these two subproblems simultaneously is shown to be NP-complete [18]. For this reason, several heuristic approaches have been developed.

One class of approaches, such as DALI [14], reduce the protein structures to some coordinate-independent space, so that they can be compared without requiring a detailed superposition. Another group of methods, such as CE [29] and MAMMOTH [23], break the proteins into short fragments, try to match the fragments from two proteins, and assemble a final alignment from matching fragment pairs. A similar approach used in SSAP [31] is to consider individual residues, and score their compatibility using inter-residue distance vectors; an alignment between two proteins is then constructed by optimizing the sum of the scores of aligned residues.

The Protein Data Bank (PDB) [4] has recently hit a milestone of 50,000 structures in April 2008. While it is possible to align two protein structures within several seconds using the pairwise alignment methods, exhaustive scanning of all the available protein structures simply becomes infeasible. This has prompted several efforts that try to quickly identify candidate structures from the database that are "worthy" of further analysis via pairwise alignment methods. These efforts can best be summarized in terms of the representation they use to capture the structural information, and the indexing method they utilize on this representation for quick retrieval.

ProGreSS [5] transforms the protein structure into a fea-

ture vector space of its curvature and torsion angles and sequence information, partitions this space into a grid and uses a voting scheme to rank the hits from this grid. [36] uses distances and angles among the secondary structure elements (SSEs) and utilizes a hashing technique to identify similar structural cores composed of triples of SSEs in two proteins. 3D-Hit [24] builds a library of short protein fragments obtained by clustering similar fragments and scans this library to identify proteins whose fragments are similar to a given query.

Similarly, [7] represents the secondary structure elements as a vector and indexes geometrical features of this vector using R*-Tree. [9] and [20] utilize geometric hashing to identify the triplets of atoms that share similar inter-residue distances with the query residues to identify all possible residue correspondences. [3] partitions the distance matrix into contact regions of the secondary structure elements and uses geometric hashing to index the distance and angle between SSEs.

There have also been several recent attempts to reduce the structural information to a sequential representation so that sequence search tools can be used directly. Protein block expert (PBE) [34] uses 16 structural motifs as a structural alphabet, whereas 3D-BLAST [33] partitions the (κ, α) dihedral angles into a 23-letter alphabet, which is then used to convert the structures into one-dimensional sequences.

We note that the structure retrieval methods surveyed only provide a coarse-level filtering of protein structures, and do not provide the sensitivity to correctly identify some of the important structural similarities. Specifically, most of these methods simply use some form of the backbone dihedral angles and thus fail to detect non-local interactions and topological similarities in related proteins [5], [7], [34], [33]. Furthermore, the fact that drastically different structures can have the same ordered composition of secondary structure elements cause these methods to return many false positives. The methods that are based on geometric hashing [36], [3], [20] do detect non-local interactions, but again at the cost of generating a huge number of false positives due to the indiscriminative nature of the representation they use. Precisely for this reason, [20] proposes the geometric hashing approach only under a massively parallel environment (more than 130,000 processors); although in such a computing environment, it is not clear whether their approach provides any benefits over the crude alternative of "one pairwise alignment per processor" scheme.

It must also be noted that these structure retrieval tools do not obtain a structural alignment, and defer this task to external pairwise alignment tools. Therefore, we really do not have the equivalent of BLAST [1] sequence search tool for structures. The strategy used in BLAST to identify similar proteins, at the same time produces their sequence alignments; on the other hand, the structure retrieval methods hitherto proposed cannot produce structural alignments due to the highly approximated and inaccurate representations and comparison schemes they utilize in favor of speed.

In this study, we present a residue-contacts indexing approach that provides sensitive and efficient retrieval of similar protein structures. Furthermore, the retrieval process we employ inherently produces residue correspondences amenable to high quality structural superposition. To the best of our knowledge, this is the first time a protein structure retrieval tool offers at the same time high quality structural alignments. The benefits of our approach is demonstrated by comparison with both structure retrieval and pairwise structure alignment tools.

II. Methods

A. Overview of the Approach

For each protein in a database of protein structures, we first extract the contact environments for each residue. Accurate comparison of the residue contacts is accomplished using their alignment with respect to a *secondary-structure enriched* amino acid substitution matrix. The metricity of the substitution matrix we have developed and the metric-preserving property of our comparison function have allowed distance-based indexing of the residue contacts. The residue contacts from the database that are similar to those of a query are efficiently retrieved using the distance-based index structure. These residue contacts are used as seeds for extension, in order to obtain high scoring segment pairs (HSPs). The HSPs are then used to induce high quality structural superpositions using a fast-converging iterative optimization procedure. We now describe each of these steps in further detail.

B. Representing the Residue Contacts

We represent each amino acid residue by the location of its C_α atom, as per convention. Although there are a number of different ways of obtaining residue-residue contacts, we use Delaunay tessellation [10], which has previously been successfully applied to packing analysis [25], protein folding [13], and structural motif mining [26].

The region of space around each residue that is closer to the enclosed residue than any other residue defines a Voronoi polyhedron. The Delaunay tessellation is then derived by connecting residues that share a Voronoi boundary. Figure 1 shows the Delaunay tessellation for a short segment of the CheY protein 1jbe. Besides associating neighboring residues, Delaunay-based definition of contacts encodes much of the proximity information and provides an abstract representation of the underlying geometry around each residue.

We reduce a residue and its contacts to a sequential *contact string* representation by ordering its contacts as they appear in the primary sequence. In order to capture both structural and biochemical information, each contact residue is further annotated with both its amino acid type and its secondary structure state. For example, the contact string for D13 (Asp13) is denoted as:

$$V_E D_C D_C^{\#} F_C S_H M_H E_C$$

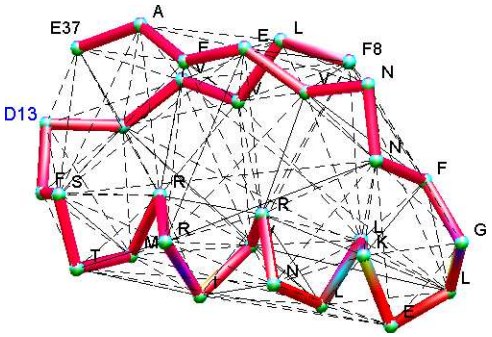

Fig. 1. Delaunay tesselation of a backbone segment (Phe8–Glu37) from the protein 1jbe. This segment contains two beta strands and one alpha helix.

where the central Asp residue is marked a pound character ("#"), and the secondary structure states are given in subscript notation. The secondary structure states consisting of alpha helices (H), beta sheets (E), and turns (C) are obtained using Dssp [16].

The comparison of two contact strings is performed in a piecewise fashion: the residues preceding and following the central residues are aligned separately. Table I shows the piecewise alignment of the contact string of Asp13 from protein 1jbe, and that of Asp21 from protein 1s8n. Individual residues are compared with respect to a secondary structure enriched amino acid substitution matrix, which is discussed next.

TABLE I

PIECEWISE ALIGNMENT OF TWO CONTACT STRINGS

1jbe Asp13:	V_E	D_C	$\Vert$	$D_C^{\#}$	$\Vert$	F_C	S_H	M_H	–	E_C	–
1s8n Asp21:	E_C	–	$\Vert$	$D_C^{\#}$	$\Vert$	E_C	A_H	R_H	G_C	D_C	G_H

C. Metric SSE-enriched amino acid substitution matrix

It has been proven that if a metric substitution matrix is used in alignment of two sequences, the alignment score also forms a metric [28]. Since our goal is to accurately and efficiently index and retrieve contact strings using distance-based indexing; we seek a substitution matrix that not only compares both SSE and sequence information, but also satisfies the metric properties. In the following, we first prove that weighted composition of two metric functions is also metric, and use this property to derive a metric SSE-enriched substitution matrix.

Definition 1: A function is said to be *metric* if it is *positive* ($f \geq 0$), *definite* ($f(x,y) = 0$ iff $x = y$), *symmetric* ($f(x,y) = f(y,x)$), and if it satisfies the *triangle inequality* ($f(x,z) \leq f(x,y) + f(y,z)$).

Theorem 1: A positive-weighted combination of two metric functions is also metric.

Proof: *Let $h = w_1 f + w_2 g$ be such a function, where f, g are metric, and w_1, w_2 are positive weights. Then h is:*

- *positive:* $h = w_1 f + w_2 g \geq w_1 0 + w_2 0 \geq 0$,
- *definite:*
  $h(x,y) = 0 \Leftrightarrow w_1 f(x,y)=0,\ w_2 g(x,y)=0 \Leftrightarrow x = y$,
- *symmetric:*
  $$\begin{aligned} h(x,y) &= w_1 f(x,y) + w_2 g(x,y) \\ &= w_1 f(y,x) + w_2 g(y,x) = h(y,x) \end{aligned}$$
- *and satisfies the triangle inequality:*
  $$\begin{aligned} h(x,z) &= w_1 f(x,z) + w_2 g(x,z) \\ &\leq w_1(f(x,y) + f(y,z)) + w_2(g(x,y) + g(y,z)) \\ &\leq h(x,y) + h(y,z) \end{aligned}$$

Therefore, h is also metric. □

Based on Theorem 1, we can construct a metric SSE-enriched substitution matrix M from metric amino acid and SSE substitution matrices AA and SS:

$$M = w_1 AA + w_2 SS$$

where w_1 and w_2 are weights adjusting the contributions of sequence and secondary structure information. For AA, we used a biologically sensitive metric matrix that we have previously developed from four-body contact propensities of amino acids [27]. SS was derived by the *inter-row distance* method [38] from the SSE substitution matrix by [35].

D. Indexing Contact Strings

We define the distance between two contact strings x and y to be the sum of the piecewise edit distances (as in Table I). The edit distance for each part of the contact strings (left, center, right) are calculated using global alignment with linear gap penalty [22]. The piecewise consideration of the contact residues around the central residue has two main advantages over a single alignment of the contact strings as employed in [6]. Firstly, it inherently enforces alignment of the central residues; this is important, because we are comparing contact strings only to quantify the feasibility of aligning the central residues. Secondly, it explores only half of the dynamic programming table, which results in twice as fast comparison of the contact strings.

Because we use a metric matrix to compare contact strings, the resulting distance measure is also metric [28]. This allows the use of distance-based indexing for efficient similarity search of contact strings. Although any distance-based indexing method can be used here, we implemented the Slim-Tree method [32] in this study. The basic idea in distance-based indexing is to hierarchically partition the data based on the given distance function, such that during a search for data that are similar to a query, the triangle inequality can be used to prune the partitions whose representatives are too dissimilar to the query, relieving the need to compare with the rest of the data in that partition. Please refer to [30] for a review of the distance-based indexing methods.

E. High Scoring Segment Pairs (HSPs)

For a given query, its contact strings are searched in the index structure to collect contact strings that are similar, within a distance threshold, to the query contact strings. Each of the resulting contact strings defines a pairing between a residue of the query, and a residue of one of the database

proteins. These pairings provide seeds that can be extended to generate segment alignments, similar to the extension phase performed in BLAST [2]. However, we introduce several notable enhancements over the basic extension scheme, as detailed next.

The contact hits to a database protein A are first sorted based on their distances to the query contact strings, such that highly similar seeds, which are more likely to be part of the final alignment path, are considered first. Starting from a seed pairing, the dynamic programming table is explored in both backward and forward directions such that a new cell is pursued further only if its alignment score is not below a certain fraction of the maximum alignment score found thus far (see Figure 2 for an illustration). The induced alignment path for a seed extension is kept if its alignment score is greater than a given threshold, and is denoted as a high scoring segment pair (HSP).

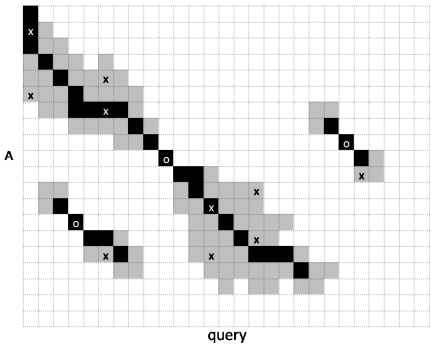

Fig. 2. Illustration of the hit extension phase to obtain HSPs from the contact string hits from a database protein A. The seeds being extended are marked with "o", and those that are pruned are marked with "x". The gray area represents the cells that are explored by the dynamic programming and the black cells form the alignment paths of the HSPs.

Naturally, we would expect multiple seed pairings along the alignment path; extending each of these seeds would simply be redundant. We therefore skip a seed if its corresponding cell has already been explored during the extension of a previous, higher scoring seed, and thereby avoid redundant seed extensions. Furthermore, we merge an HSP if its alignment path intersects with that of a previous HSP, which results in construction of longer alignments and thus avoids generation of multiple short alignments, which is a common problem of BLAST-like approaches. On randomized searches in ASTRAL-25 database [8], these heuristics eliminated more than 40% of the seeds that were redundant, and merged 7% of the HSPs which otherwise would have been produced as shorter, lower scoring alignments.

F. Structural Superposition

The correspondences defined by the HSPs can directly be used to calculate a structural superposition [15]. We further optimize this superposition using an iterative procedure commonly employed by pairwise structure alignment methods. From the initial superposition, a new set of correspondences is generated using dynamic programming to minimize the total distance between corresponding residues. The new set of correspondences is again used to induce a superposition, and the iteration is repeated until the translation-rotation matrix no longer changes. Because the correspondences identified by the HSPs already match structurally compatible residues, the convergence is achieved usually in only a few iterations. The final set of structurally aligned proteins are sorted based on TM-score [37], which is a normalized structural alignment score that considers both coverage and accuracy of the alignment, and is shown to be in accordance with human expert evaluations.

III. Experiments

The parameters used in RCIndex were optimized for retrieval accuracy and alignment quality on an independent training dataset (available on RCIndex web service) using the Nelder-Mead simplex method [17]. The experiments were run on a Pentium 2.6 GHz personal computer. In the following sections, we show that RCIndex is able to quickly and successfully identify similar protein structures (the retrieval problem). Then we show that in addition to successful similarity search capability, it also produces high quality structure alignments (the alignment problem).

A. Protein Structure Retrieval

The dataset we use to evaluate the retrieval task appears previously in [3]. This dataset is derived from ASTRAL-40 database [8], and consists of 10 proteins from each of the Globins family (a.1.1.2) and the Ser/Thr Kinases family (d.144.1.1), and 180 proteins from the four major SCOP classes (all α, all β, α/β, and $\alpha+\beta$) [21]. Each of the Globin and Ser/Thr Kinases proteins were used as queries against the database of 200 proteins.

TABLE II
AVERAGE ACCURACY ON THE DATABASE OF 200 PROTEINS.

	average number of retrievals required					
recall	DALI	CE	TopScan	ProtDex	ProtDex2	**RCIndex**
1	1	1	1	1	1	1
2	2	2	2	3	2	2
4	4	4	5	7	4	4
6	6	6	8	12	6	6
8	8	8	14	21	9	8
10	10	10	29	79	16	10

Table II compares the number of retrievals required for different number of correct retrievals. For each recall level, RCIndex shares the same 100% accuracy with the detailed pairwise structural alignment programs DALI and CE. Furthermore, the time spent to search the database is much better than these alignment methods, giving comparable running times as the less accurate database scanning methods TopScan [19], Protdex and Protdex2 [3] (See Table III for time comparisons).

Both Globins and Ser/Thr Kinases are highly conserved in structure across different organisms, even though their sequences may vary greatly. For example, the Globins 1ash and 3sdh have 9% sequence identity, and the Ser/Thr Kinases 1b6c:b and 1tki have 13% sequence identity. The alignments

TABLE III

RUNNING TIMES ON THE DATABASE OF 200 PROTEINS.

method	total time for 20 queries (hh:mm:ss)	average time per query (hh:mm:ss.mmmm)
DALI	52:36:08	02:37:48.40
CE	10:23:03	00:31:09.15
TopScan	00:00:59	00:00:02.95
ProtDex	00:00:43	00:00:02.15
ProtDex2	00:00:16	00:00:00.80
RCIndex†	00:03:32	00:00:16.59

†The running times for RCIndex were interpolated to the same time scale in [3] using running times of CE for normalization. The actual running time in our experimental environment was better: 10.2s per query.

obtained by RCIndex for these proteins are shown in Figure 3. Whereas the Globin domain is relatively simple in structure, composed of all alpha helices, the Ser/Thr Kinases display a more complex structure composed of alpha helices and beta sheets. RCIndex was able to provide good structural alignments ($TM\text{-}score \geq 0.67$) for all of the correct family pairs.

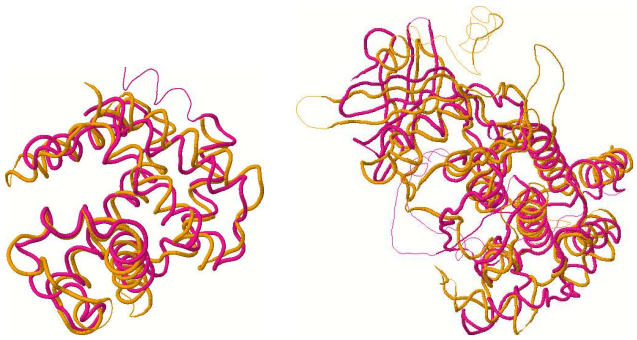

Fig. 3. Example RCIndex alignments for Globins (Left, 1ash-3sdh, TM-score=0.77) and Ser/Thr Kinases (Right, 1b6c:b-1tki, TM-score=0.67).

B. Structure Alignment

We note again that the current structure database scanning methods only identify candidate proteins, and do not provide any structural superpositions. On the other hand, RCIndex identifies the candidate proteins, and the HSP alignments produced during this retrieval process further allow us to induce structural superpositions. Even though RCIndex is able to produce structural alignments, it still needs to be established that these alignments are of reasonable quality when compared with those produced by pairwise structure alignment methods. For this purpose, we used the *10 difficult pairs* of structures [12], which have previously been used to benchmark structural alignment methods. For RCIndex, one of the proteins in each pair was used to initialize the database against which the other protein was searched. For the other methods, we used the respective web services. Example alignments produced by RCIndex are shown in Figure 4.

Structural alignments RCIndex produces were not only "reasonable", but were in fact comparable to or better than those produced by the popular pairwise structure alignment tools (see Table IV). RCIndex achieved the best average alignment quality as measured by the TM-score, with coverage (%N) comparable to that of other methods. Only SSAP produced alignments that are slightly longer than those of RCIndex; however, at the cost of significantly higher RMSD errors. One of the extreme cases is the 1ede-1crl pair, for which SSAP gives the largest RMSD error of any of the alignments by any method.

Vorolign [6], which is a pairwise structure alignment method also based on Voronoi contacts, gives the best RMSD, at the cost of significantly shorter alignments, and lower TM-scores. Furthermore, Vorolign fails to produce an alignment for 1ten-3hhrb pair. We attribute the differences between the alignment qualities of Vorolign and RCIndex to the sensitivity of the metric substitution matrix we have developed, and to the more accurate distance function we use to compare contact strings.

Fig. 4. Example RCIndex alignments from 10 difficult pairs. Left: 1ubq and 1fxia are from different superfamilies in the Ubiquitin-like beta grasp fold. Right: 3hlab and 2rhe are from different families of the Immunoglobulin superfamily, and have the lowest sequence identity (4%) among the 10 difficult pairs.

IV. DISCUSSION

We have presented a protein structure retrieval method which achieves accurate identification of similar proteins from a database of proteins. More remarkably, it at the same time produces structural alignments that are comparable to or better than those produced by current pairwise structural alignment tools. The success of our approach is based on a sensitive representation and a metric comparison of the residue contacts. The observation that similar proteins share similar residue contacts is exploited in developing a *hit & extend* methodology where similar residue contacts are quickly identified with the help of distance-based indexing, and extended to obtain high scoring segment pairs (HSPs). The HSPs derived in this way inherently contain correspondences between structurally compatible residues, and provide a good basis for an iterative optimization of structural superposition.

While RCIndex is presented as a specific implementation, it directs to a more general, extensible framework of structural search and alignment. Particularly, different substitution matrices or distance functions that incorporate geometrical or biochemical nature of the residue environments can be developed and used in RCIndex without any changes to the rest of the algorithm, provided that they satisfy metric properties, or permit other efficient indexing strategies. The extension phase of RCIndex can also incorporate other filters

TABLE IV

COMPARISON OF ALIGNMENT QUALITY ON 10 DIFFICULT PAIRS.

				CE			SSAP			DaliLite			Vorolign*			RCIndex		
		avg. size	%identity	rmsd	%N	TM	rmsd	%N	TM	rmsd	%N	TM	rmsd	%N	TM	rmsd	%N	TM
1ubq	1fxia	86	7	3.82	0.84	0.49	4.02	0.88	0.49	2.69	0.80	0.53	2.16	0.42	0.40	2.48	0.82	0.56
1ten	3hhrb	143	19	1.90	0.97	0.80	1.88	0.96	0.81	1.91	0.96	0.79	—	—	—	1.74	0.97	0.82
3hlab	2rhe	106	4	3.38	0.85	0.51	4.99	0.85	0.45	3.03	0.76	0.51	2.18	0.39	0.43	3.15	0.84	0.54
1paz	2azaa	124	12	2.86	0.70	0.52	3.41	0.73	0.53	2.46	0.68	0.53	2.26	0.59	0.63	2.73	0.71	0.54
1mola	1cewi	101	15	2.34	0.86	0.66	2.46	0.87	0.66	2.26	0.86	0.67	1.99	0.76	0.70	2.12	0.86	0.68
2rhe	1cid	146	11	2.91	0.85	0.64	3.72	0.90	0.64	3.02	0.83	0.63	1.95	0.58	0.62	2.79	0.87	0.67
1ede	1crl	422	5	3.85	0.71	0.57	9.25	0.86	0.46	3.43	0.67	0.56	3.22	0.34	0.44	4.86	0.76	0.56
2sim	1nsba	386	8	2.97	0.72	0.63	5.42	0.84	0.65	3.28	0.76	0.65	2.63	0.52	0.71	3.67	0.81	0.68
2gmfa	1bgeb	140	13	4.64	0.90	0.52	5.36	0.97	0.56	3.20	0.77	0.56	2.17	0.46	0.50	3.86	0.91	0.61
4fgf	1tie	145	10	3.04	0.94	0.70	3.23	0.96	0.69	2.88	0.90	0.70	2.00	0.59	0.63	2.79	0.94	0.72
average:		180	10	3.17	0.83	0.60	4.37	0.88	0.59	2.82	0.80	0.61	2.28	0.52	0.56	3.02	0.85	0.65

* Vorolign reports alignments for multiple substitution matrices; here we use the SM-THREADER matrix [11], which gives the best results.

for candidate evaluation, or other structural compatibility functions. We are currently evaluating such extensions that can further increase the efficiency of RCIndex.

Our current focus is on demonstrating the benefits of RCIndex on large-scale experiments and extending it to structural motif mining applications. Initial results (not shown here) indicate that RCIndex is scalable to very large protein structure databases and still gives time performance comparable to that of less accurate coarse-level scanning methods, and accuracy comparable to detailed structure alignment methods. For the large non-redundant database of more than 4,000 proteins currently served by the RCIndex web service, a typical search takes around 1 minute to perform, including structural alignment of the top scoring hits. This is a remarkable saving over exhaustive scanning by pairwise alignment tools, which typically takes days to weeks to complete for the same database.

V. ACKNOWLEDGEMENTS

This research was partially supported by US National Science Foundation (NSF) Grants IIS-0546713 and DBI-0750891; and Turkish Scientific and Research Council (TUBITAK) Grant 107E173.

REFERENCES

[1] S. Altschul, W. Gish, W. Miller, E. Myers, and D. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, 1990.

[2] S. Altschul, T. Madden, A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. Lipman. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*, 25:3389–3402, 1997.

[3] Z. Aung and K. Tan. Rapid 3D protein structure database searching using information retrieval techniques. *Bioinformatics*, 20:1045–1052, 2004.

[4] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. Shindyalov, and P. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28:235–242, 2000.

[5] A. Bhattacharya, T. Can, T. Kahveci, A. K. Singh, and Y.-F. Wang. Progress: Simultaneous searching of protein databases by sequence and structure. *Pacific Symposium on Biocomputing*, 9:264–275, 2004.

[6] F. Birzele, J. E. Gewehr, G. Csaba, and R. Zimmer. Vorolign: fast structural alignment using Voronoi contacts. *Bioinformatics*, 23(2):e205–e211, 2007.

[7] O. Camoglu, T. Kahveci, and A. K. Singh. Psi: indexing protein structures for fast similarity search. *Bioinformatics*, 19:181–183, 2003.

[8] J. Chandonia, G. Hon, N. Walker, L. L. Conte, P. Koehl, M. Levitt, and S. Brenner. The ASTRAL compendium in 2004. *Nucleic Acids Research*, 32:189–192, 2004.

[9] L. Chen, R. Oughtred, H. M. Berman, , and J. Westbrook. Targetdb: a target registration database for structural genomics projects. *Bioinformatics*, 20(16):2860–2862, 2004.

[10] B. Delaunay. Sur la sphre vide. *Izvestia Akademii Nauk SSSR, Otdelenie Matematicheskikh i Estestvennykh Nauk*, 7:793–800, 1934.

[11] Z. Dosztanyi and A. Torda. Amino acid similarity matrices based on force fields. *Bioinformatics*, 17:686–699, 2001.

[12] D. Fischer, A. Elofsson, D. Rice, and D. Eisenberg. Assessing the performance of fold recognition methods by means of a comprehensive benchmark. *Pacific Symposium on Biocomputing*, pages 300–318, 1996.

[13] H. Gan, A. Tropsha, and T. Schlick. Lattice protein folding with two and four-body statistical potentials. *Proteins*, 43:161–174, 2001.

[14] L. Holm and C. Sander. Protein structure comparison by alignment of distance matrices. *J. Mol. Biol.*, 233:123–138, 1993.

[15] W. Kabsch. A discussion of the solution for the best rotation to relate two sets of vectors. *Acta Crystallogr.*, A34:827–828, 1978.

[16] W. Kabsch and C. Sander. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers*, pages 2577–637, 1983.

[17] J. C. Lagarias, J. Reeds, M. H. Wright, and P. E. Wright. Convergence properties of the Nelder-Mead simplex method in low dimensions. *SIAM Journal of Optimization*, 9(1):112–147, 1998.

[18] R. Lathrop. The protein threading problem with sequence amino acid interaction preferences is NP-complete. *Protein Eng.*, pages 1059–1068, 1994.

[19] A. Martin. The ups and downs of protein topology: rapid comparison of protein structure. *Protein Eng.*, 13:829–837, 2000.

[20] T. Milledge, G. Zheng, T. Mullins, and G. Narasimhan. Sblast: Structural basic local alignment searching tools using geometric hashing. *Proceedings of the 7th IEEE International Conference on Bioinformatics and Bioengineering*, pages 1343–1347, 2007.

[21] 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. *J. Mol. Biol.*, 247:536–540, 1995.

[22] S. B. Needleman and C. D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol Biol.*, 48:443, 1970.

[23] A. R. Ortiz, C. E. Strauss, and O. Olmea. Mammoth (matching molecular models obtained from theory): An automated method for model comparison. *Protein Sci*, 11(11):2606–2621, 2002.

[24] D. Plewczynski, J. Pas, M. von Grotthuss, , and L. Rychlewski. 3d-hit: fast structural comparison of proteins. *Appl. Bioinformatics*, 1(4):233–235, 2002.

[25] F. Richards. The interpretation of protein structures: total volume, group volume distributions and packing density. *J. Mol. Biol.*, 82:1–14, 1974.

[26] A. Sacan, O. Ozturk, H. Ferhatosmanoglu, and Y. Wang. Lfm-pro: A tool for detecting significant local structural sites in proteins. *Bioinformatics*, 2007.

[27] A. Sacan and I. H. Toroslu. Amino acid substitution matrices based on 4-body Delaunay contact profiles. *IEEE 7th Intl Symp on*

Bioinformatics and Bioengineering (IEEE-BIBE2007), pages 796–802, 2007.

[28] P. Sellers. On the theory and computation of evolutionary distances. *J. Appl. Math. (SIAM)*, 26:787–793, 1974.

[29] I. N. Shindyalov and P. E. Bourne. Protein structure alignment by incremental combinatorial extension (CE) of optimal path. *Protein Engineering*, 11(9):739–747, 1998.

[30] M. Taskin and Z. M. Ozsoyoglu. Improvements in distance-based indexing. *Proceedings of the 16th International Conference on Scientific and Statistical Database Management, SSDBM'04*, pages 161–170, 2004.

[31] W. Taylor and C. Orengo. Protein structure alignment. *J. Mol. Biol.*, 208(1):1–22, 1989.

[32] J. C. Traina, A. J. M. Traina, B. Seeger, and C. Faloutsos. Slim-trees: High performance metric trees minimizing overlap between nodes. In *Proc. of the 7th Intl. Conf. on Extending Database Techn.*, pages 51–65, 2000.

[33] C.-H. Tung, J.-W. Huang, and J.-M. Yang. Kappa-alpha plot derived structural alphabet and BLOSUM-like substitution matrix for rapid search of protein structure database. *Genome Biology*, 8:R31.1–R31.16, 2007.

[34] M. Tyagi, P. Sharma, C. S. Swamy, F. Cadet, N. Srinivasan, A. G. de Brevern, , and B. Offmann. Protein block expert (pbe): a web-based protein structure analysis server using a structural alphabet. *Nucl. Acids. Res.*, 34:W119–W123, 2006.

[35] A. Wallqvist, Y. Fukunishi, L. R. Murphy, A. Fadel, and R. M. Levy. Iterative sequence/secondary structure search for protein homologs. *Bioinformatics*, 16:988–1002, 2000.

[36] M. M. Young, A. G. Skillman, and I. D. Kuntz. A rapid method for exploring the protein structure universe. *Proteins*, 34(3):317–32, 1999.

[37] Y. Zhang and J. Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins*, 57:702–710, 2004.

[38] E. Zintzaras. A comparison of amino acid distance measures using procrustes analysis. *Computers in Biology and Medicine*, 29(5):283–288, 1999.