Fast Detection of Common Geometric Substructure in Proteins

L. Paul Chew * Klara Kedem † Jon Kleinberg ‡ Dan Huttenlocher §

July 7, 1998

Abstract

We consider the problem of identifying common three-dimensional substructures between proteins. Our method is based on comparing the shape of the α-carbon backbone structures of the proteins in order to find 3D rigid motions that bring portions of the geometric structures into correspondence. We propose a geometric representation of protein backbone chains that is compact yet allows for similarity measures that are sensitive to noise and outliers. We represents the structure of the backbone as a sequence of unit vectors, defined by each adjacent pair of α-carbons; we then define a measure of the similarity of two protein structures based on the RMS (root mean squared) distance between corresponding orientation vectors in the two proteins.

Our measure has several advantages over standard position-based RMS measures that are commonly used for comparing protein shapes. In particular, the measure behaves well for comparing substructures, because unlike position-based measures the nonmatching portions of the structure do not dominate the measure. At the same time, it avoids the quadratic space and computational difficulties associated with the use of distance matrices and contact maps. We show applications of our approach to detecting common contiguous substructures in pairs of proteins, as well as the more difficult problem of identifying common protein domains (i.e., larger substructures that are not necessarily contiguous along the protein chain).

*Supported in part by the National Science Foundation (EIA-9726388), by DARPA through ONR (N00014-96-1-0699), and by a grant from the US-Israeli Binational Science Foundation.

†Supported in part by a grant from the Israel Science Foundation, by the Mary Upson Award from the College of Engineering at Cornell University, and by a grant from the US-Israeli Binational Science Foundation.

‡Supported in part by an Alfred P. Sloan Research Fellowship and by NSF Faculty Early Career Development Award CCR-9701399.

§Supported in part by a grant from the US-Israeli Binational Science Foundation.

1 Introduction

As an increasing number of protein structures become known, the need for algorithms to analyze three-dimensional conformational structure increases as well. The search for common substructure is of value in uncovering relationships among different proteins — for inferring similarities in function and discovering common evolutionary origins. There is now widespread agreement that similarities among distantly related proteins are often preserved at the level of three-dimensional structure, even after very little similarity remains at the sequence level [7]. We are interested in developing efficient comparison methods for three-dimensional protein structures.

The goal of our work is to automatically identify common substructures shared by two proteins, based on similarity in their three-dimensional structure. This is a problem that has recently received considerable attention in the biological research literature (cf. [13]). For protein chains that are very similar (e.g., candidate structures produced in crystallography experiments), a common measure of similarity is the RMS (root mean square) distance between the positions of corresponding atoms of the proteins, after the proteins have been superimposed by a three-dimensional rigid motion. While this approach is reasonable for comparing two structures that are highly similar, it does not work well for finding similar substructures in otherwise dissimilar proteins. The problem with dissimilar proteins is that the portions of the structure that do not match dominate the RMS value, making it difficult to identify matching substructures. This is an instance of the classical issue of outliers, or bad data, dominating the measure in fitting or estimation problems.

The limitations of RMS matching have been addressed by a number of means, the most common of which are based on comparing the distance matrices (or contact maps) of the two proteins (e.g., [16, 15, 9]). The distance matrix records the pairwise distances between the atoms within a given protein (e.g., for all the α-carbons of the residues). For a protein with n atoms the distance matrix contains $O(n^2)$ entries. Note that the matrix is symmetric, and thus only the upper or lower triangular portion is generally used.

The distance matrix representation is invariant with respect to the position and orientation of the protein, and can thus be compared directly for two proteins, without the need to find a rigid motion superimposing two proteins as with RMS. Two identical proteins will have identical distance matrices, because all the pairwise distances are the same. The main drawback of the distance matrix is that it leads to relatively computationally expensive algorithms, particularly for identifying substructures. Using the distance matrix representation of two proteins, a contiguous matching substructure corresponds to submatrices of the two distance matrices that are nearly the same. Finding a match that consists of several such subsequences involves not only finding several such blocks, but further checking that the corresponding substructures can be simultaneously aligned. These problems are generally solved using optimization techniques (e.g., [7, 9]).

In this paper we present a new algorithm for the efficient detection of common substructure in proteins, based on a representation of protein chains different from both three-dimensional coordinates and distance matrices. We begin from the standard representation of a protein by the sequence of α-carbons that form its backbone chain. Rather than encoding the coordinates of these atoms, however, we use an orientation-based representation: for each successive pair of atoms along the backbone chain, we record the unit vector in the direction from α-carbon i to α-carbon $i+1$. Note that because the separation of these successive atoms is essentially a unit distance (approximately four angstroms), this representation captures the structure of the backbone. As such, it is much more compact than the distance matrix or contact map ($O(n)$ versus $O(n^2)$), and we show below that it is much less sensitive to noise than the coordinates of the atoms themselves.

Our algorithm based on this representation consists of the following three steps.

1. Determine a small set of *shifts* or *contiguous alignments* that are likely to include those alignments that bring matching substructures into correspondence.

2. For each shift chosen above, determine the contiguous substructures that can be *superimposed* (i.e., those that can be geometrically aligned using a 3D rigid motion).

3. Combine the contiguous substructures chosen above to identify common protein *domains* (larger substructures that are not necessarily contiguous along the protein chain) that can be *superimposed.*

The first step of our algorithm runs in $O(n \log n)$ time, where n is the number of α-carbons in the larger of the two proteins being compared. The second step takes $O(n)$ time for each shift identified in step one; typically only the best 10 shifts provide any useful information, but our current implementation checks the best 20. The third step takes time $O(s^2)$ where s is the number of substructures found in step two.

This problem inherently involves several types of *alignment* and some care must be taken to ensure that the terminology is understandable. The first step involves the alignment of two sequences by a simple shift of one relative to the other. That is, for integers $k = 0, 1, \ldots, n-1$, we match each residue i of the shorter protein with residue $i+k$ of the longer protein. Here addition is taken modulo n, so that our shift includes a "wrap-around". The wrap-around ensures that each unit vector in protein A is compared against each unit vector in protein B; of course, substructures that wrap the from the end of a protein back to the beginning are meaningless. Note that this type of shift-based alignment is different from, and simpler to work with than, the standard biological notion of *sequence alignment*, which allows for insertions and deletions into a sequence. We refer to our type of alignment as a *contiguous alignment* or *shift*. The second step involves geometrically aligning two substructures using a 3D rigid motion; we use the term *superimpose* when referring to this type of (geometric) alignment.

Note that using our unit-vector representation, two structures that are the same except for a bend in the middle will have different orientation vectors for all the α-carbons after the bend. This is in contrast to the standard (ϕ, ψ) angle representation that has been used in other work on protein structure comparison ([13]). Any (ϕ, ψ)-based technique is using a local coordinate system; in our single-bend example, the use of a local coordinate system implies that all the orientations match except for those directly at the bend. Our unit-vector representation more closely reflects the 3D positions of the α-carbons.

Our method uses a variant of the RMS distance, based on measuring differences between the orientation vectors at corresponding α-carbons rather than measuring differences between their positions. The key underlying observation is that the difference between two orientation vectors is bounded by a small constant, whereas the distance between two points can grow arbitrarily large. Thus the *orientation RMS* that we propose is naturally robust with respect to the differences between nonmatching parts of the structure whereas the conventional *position RMS* is not.

The RMS distance. A basic method for determining geometric similarity between two sets of points is the *root mean square* (RMS) distance on point sets. This measure of distance requires a one-to-one correspondence between the point sets. The standard (position-based) RMS distance between point sets $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$ is defined as the square root of the sum of the squared distances between the points of A and the corresponding points of B. Often we want the best possible RMS distance; that is, we want the RMS distance between A' and B where A' is the set A after application of the 3D rigid motion that minimizes the RMS distance. We refer to this as the *minimum RMS distance* between A and B.

To compute the minimum RMS distance, we first transform A to make it have the same center-of-mass as B. Then we compute the *correlation matrix* for A and B. This is a 3×3 matrix in which the p, q entry is $\sum_{i=1}^{n} a_{i,p} b_{i,q}$. This matrix is factored using the singular value decomposition; the resulting factors can be used to construct in constant time both (1) the orthogonal matrix that brings A into the closest possible match with B and (2) the resulting RMS distance. For a concise description of the algorithm and an explanation of why it works see [6] (where it is referred to as the orthogonal Procrustes problem) or [10].

The minimum RMS distance between two sets of atoms in $\Re^3$ is useful primarily when aligning two structures that are extremely similar. Thus, it is widely used for aligning structures produced in the course of crystallographic or NMR analysis of molecules.

It is less appropriate for comparing structures that have only a distant similarity. This is due to its sensitivity to outliers. Indeed, a few outliers can have a significant effect on the resulting minimum RMS distance. For instance, consider sets A and B, each with n points, where the minimum RMS distance between A and B is 0. Now suppose we take the point b_1 and gradually move it along a straight line. As we move b_1, the minimum RMS distance between A and B increases without bound, and the rigid motion corresponding to this minimum distance also changes so that the congruent point sets $A - \{a_1\}$ and $B - \{b_1\}$ become arbitrarily far apart. Thus a small number of outliers can hide a large shared substructure.

Another issue with the minimum RMS distance is that points far from the center of mass are, in effect, weighted more heavily than points near the center of mass. Consequently, a common "core" shared by two structures is difficult to detect using the minimum RMS distance. This is problematic in the case of proteins, where some crucial similarities occur near the center of mass.

Our Approach. Our goal is to develop an approach that avoids the drawbacks discussed above while retaining the computational efficiency of the minimum RMS calculations. The underlying idea is to use an RMS distance based on orientation instead of position. Consider a protein chain as a sequence of vectors; each vector goes from one α-carbon to the next. Each such vector has an orientation, represented by the corresponding unit vector. We view a protein as a sequence of unit vectors. By chaining the unit vectors end to end, we obtain the standard model of a protein as a sequence of α-carbons in space. Alternatively, we can place all of the unit vectors at the origin; we then view a protein as a sequence of points on the unit sphere.

We compare two proteins by comparing these sequences of points on the unit sphere. In particular, we make use of the minimum RMS distance between sequences of unit vectors. Note that the unit vectors all have a common origin before we start, so we do not have to do the center-of-mass calculation that is needed for the standard, position-based, minimum RMS distance. We use the term *position-RMS* to refer to the standard, position-based, minimum RMS distance. We use the term *orientation-RMS* to refer to the minimum RMS distance calculated from a sequence of unit vectors derived from our orientation-based representation.

Because the maximum squared distance between any two unit vectors is 4, no single outlier can have a large effect on the orientation-RMS. As a result, it exhibits a natural resilience to the presence of a small number of outliers.

Related Work. A number of different methods for computing structural similarity have been proposed. See [2, 8, 13] for recent surveys.

Some of the approaches are based on sequence alignment extended to handle pairwise relationships among residue positions (Taylor and Orengo [14], Orengo et al. [11], Sali and Blundell [12]). There has also been work on extending sequence-based approaches to handle "out-of-order" alignments between two sequences (Vriend and Sander [15], Holm and Sander [9], Grindley et al. [5]). A variant of the *contact map* is often used for this purpose, based on the pairwise distances between α-carbons, or the spatial relationships among secondary structural elements. In particular, Vreind and Sander [15] use the distance matrix to find pairs of matching fragments of the two proteins and then combine them into larger units which are compatible in $3D$. The latter step is based on comparisons of the centers of mass of the fragments and of relative rotations between combined fragments.

Fischer et al [4] apply *geometric hashing* to find matching pairs of atoms (alpha carbons) between two molecules. The common subsequences they find are not necessarily consecutive. To use geometric hashing, a *base molecule* is preprocessed to find all possible matching coordinate frames. This may involve up to $O(n^3)$ pre-processing and storage, since a rigid motion in three dimensions is specified by its action on three points. (Distance constraints imposed by the protein can improve this bound somewhat.) Given a second *target molecule*, each triplet of points "votes" for a rigid motion with respect to the base molecule; a large number of common votes indicates a possible large match between the two molecules. Assuming that the rigid motions are indexed in a well-distributed hash table, the running time is $O(n^3)$. Recent unpublished work on this technique has suggested that it may be possible to reduce the running time for protein matching applications to a roughly quadratic bound.

Finally, our approach may bear the most similarity to a computational technique of Yee and Dill [16]. They also search over contiguous alignments of two protein sequences to discover similarities, but their notion of *similarity* is different from ours. It is based on shifting the alignments of the distance matrices of the two proteins and computing the "distance" between the matrices (by means of the Frobenius norm). The shifts that give the smallest norms are predicted to yield matching contiguous alignments. Though the running time is not discussed explicitly in their work, it is easy to see that it can be $O(n^3)$. (It is possible that an application of the Fast Fourier Transform could improve this to $O(n^2 \log n)$.)

2 Fast Substructure Detection

Consider the case in which two proteins contain *contiguous* substructures that are approximately similar. By *contiguous*, we mean a portion of structure corresponding to an unbroken substring of the protein's primary sequence. We first compute the orientation-RMS, as defined above, for each relative shift of the two protein sequences. Using the FFT and assuming each protein has length bounded by n, this can be accomplished in $O(n \log n)$ time rather than the cruder bound of $O(n^2)$. When we reach the shift at which there is a substructure match, we expect the orientation-RMS to drop considerably. By looking for shifts at which there is an unusually low orientation-RMS, we can detect approximate common substructures.

Why should we expect a drop of this sort? It is here that the use of orientation-RMS, rather than position-RMS, is crucial. Consider that a "bad" match for orientation-based matching contributes at most 4 to the sum of squared distances; a single bad match in the position-RMS can have an unbounded effect. Thus, each outlier in the orientation-based setting contributes at most a constant amount to the error, and thus cannot prevent a good matching region from standing out.

Intuitively, two unrelated proteins "look random" to each other. To make this intuition more precise, consider two proteins placed at random orientations with respect to each other. We assume that each of these two unrelated proteins can be approximated by a sequence of randomly chosen unit vectors.

Lemma 1 *The expected (nonminimized) RMS distance between two sequences of randomly chosen unit vectors in* $\Re^3$ *is* $\sqrt{2}$.

Proof: By symmetry, we may assume that one of the vectors is $y = (0, 0, 1)$. For a unit vector x, let $f(x)$ denote the result of reflecting x through the xy-plane. If we generate a unit vector x from the uniform distribution on the sphere, and then output x or $f(x)$ with equal probability, the result is still uniformly distributed. But

$$(x - y)^2 + (f(x) - y)^2 = (x - y)^2 + (x + y)^2 = 2x^2 + 2y^2 = 4,$$

and so the expected squared distance to y is 2. □

Experiment confirms that this lemma holds when applied to proteins. This is in spite of the obvious fact that proteins are not random sequences of unit vectors; indeed, adjacent vectors within a protein are highly correlated. For instance, the angles between adjacent vectors are highly constrained and there are larger substructures that occur with high frequency (e.g., alpha helices and beta sheets). The *Ramachandran plot* of angles between adjacent residues [3] depicts exactly the correlations that occur in typical secondary structural units. Though adjacent vectors within a protein are not random it appears that the intuition that unrelated proteins "look random" to each other is valid.

One intuitively expects that the orientation-RMS (the *minimized* RMS distance between sets of random unit vectors) should achieve a value not much below $\sqrt{2}$. To achieve a more precise bound, we analyze the correlation matrix used during the RMS calculation.

The minimized RMS distance between two sequences of unit vectors, each of length n, is

$$\sqrt{\frac{2n - 2 * \mathrm{trace}(\mathrm{svd}(A))}{n}}$$

where A is the correlation matrix and svd(A) is the diagonal matrix in the SVD decomposition of A [6]. For unit vectors picked uniformly in the unit sphere (say (x_1, x_2, x_3) from the first protein and (y_1, y_2, y_3) from the second) we need the distribution for x_1y_1, x_1y_2, etc. These are the terms that appear in the correlation matrix. These terms all have the same probability distribution, so we analyze the random variable X_1Y_1.

For x_1y_1, we can assume that x_1 and y_1 are independent and uniformly distributed in the interval $[-1, 1]$. This follows from the correspondence between uniform density on a sphere's surface and uniform density on the sphere's diameter.

The random variable $X = X_1Y_1$ is not uniform, but we're adding a large number of such random values (one for each unit vector in the protein chain), so, once we have the mean and variance, we should be able to use the Central Limit Theorem.

$$E[X] = \frac{1}{4}\int_{-1}^{1}\int_{-1}^{1} x_1y_1 dx_1 dy_1 = 0$$

$$E[X^2] = \frac{1}{4}\int_{-1}^{1}\int_{-1}^{1} x_1^2y_1^2 dx_1 dy_1 = \frac{1}{9}$$

Thus, the standard deviation is $\sigma = \frac{1}{3}$.

n	$D_{formula}$	$D_{experiment}$
100	1.310	1.311
200	1.341	1.341
300	1.355	1.354
400	1.363	1.363
500	1.369	1.368

Table 1: The expected orientation-RMS for different length sequences of random unit vectors. The formula is derived in the text. The experiments used sequences of random unit vectors generated using MATLAB. For each length, the experimental value shown is the mean for 1000 trials.

Applying the Central Limit Theorem, we conclude that each individual entry in the correlation matrix can be represented by a random variable of the form $\frac{\sqrt{n}}{3}\eta(0,1)$ where $\eta(0,1)$ represents the normal distribution with mean 0 and standard deviation 1. Unfortunately, this isn't enough to determine the final form of the correlation matrix: each individual entry has this distribution, but the entries are not independent. Still, an analysis under the (false) assumption of independence may be suggestive of how the matrices behave in practice; we confirm this with experimental results.

Let B be a 3×3 matrix with entries chosen independently from $\eta(0,1)$. We need to determine expected values for the trace of the SVD of this matrix. It's a simple matter to run 10,000 experiments of this type using MATLAB. Experimentally, the desired trace has mean 4.26 with standard deviation 1.06.

The analysis above indicates that trace(svd(A)) in the formula for distance can be replaced by its expected value which is $\frac{\sqrt{n}}{3}b$ where b is the expected value of trace(svd(B)). Substituting our experimentally derived value for b, the expected distance D is $\sqrt{2 - \frac{2.84}{\sqrt{n}}}$. This is the approximate distance that we should see between two unrelated proteins after the proteins have been rotated to agree as much as possible.

Table 1 shows values for the expected value of the orientation-RMS using (1) the formla derived above and (2) experimental data using sequences of random unit vectors. The formula and experiments agree more completely than we had expected. Apparently, the (false) assumption of independence that we used above still allowed the derivation of a valid formula.

These results agree quite well with our experiments on real proteins, although the orientation-RMS for real proteins is a bit smaller than the values derived above. See Figures 1 and 2. These figures show the orientation-RMS plotted for each possible shift. Figure 1 compares a protein of length 65 with one of length 104. The expected orientation-RMS derived using the above formula (for a comparison of length 65) is 1.284 while the figure appears to indicate an average value somewhat below this. Figure 2 compares a protein of length 382 with one of length 457. The expected orientation-RMS, by formula, is 1.362 while, again, the figure indicates a value below this. The discrepancy can be explained by the fact that our formula models proteins as sequences of random unit vectors, an assumption that is certainly untrue.

In summary, both analysis and experiment indicate that for two unrelated proteins (1) the expected (nonminimized) RMS-distance for the corresponding sequences of unit vectors is $\sqrt{2}$ when the proteins are placed at random orientations and (2) the expected orientation-RMS (i.e., the RMS-distance for unit vectors when the proteins are rotated to align as well as possible) is roughly equal to $\sqrt{2 - \frac{2.84}{\sqrt{n}}}$ where n is the length of the smaller of the two proteins.

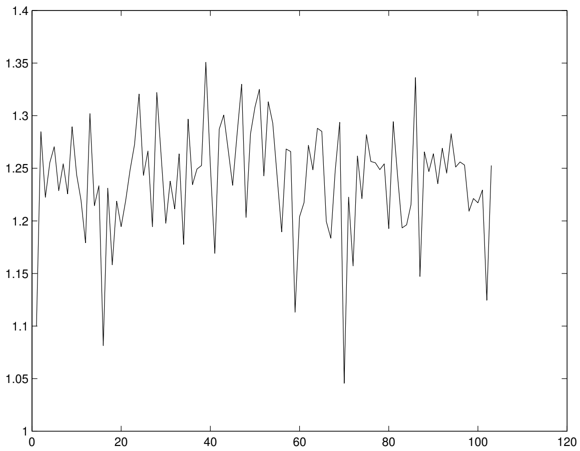

Figure 1: Orientation-RMS for each shift of 2cro and 2wrp.

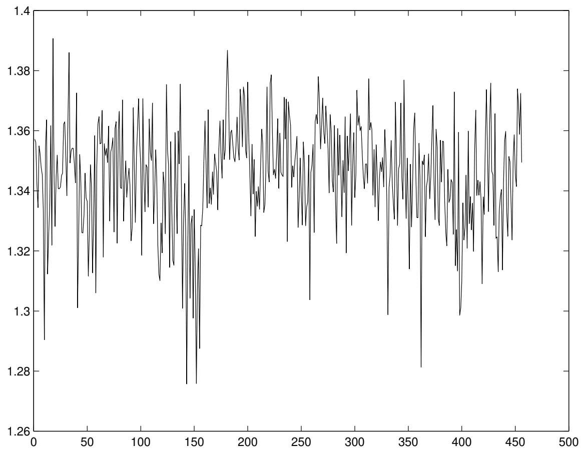

Figure 2: Orientation-RMS for each shift of 1hsc and 2yhx.

If two proteins have a substructure in common and if the protein sequences are shifted to bring the two similar substructures into correspondence then the orientation-RMS should decrease. The following lemma tries to capture this notion, building on our view of unrelated portions of protein chains as looking "random" with respect to one another.

Lemma 2 *Let $v_1, \ldots, v_k$ be unit vectors in $\Re^3$, and let $a_1, \ldots, a_n, b_1, \ldots, b_n$ be unit vectors chosen uniformly and independently at random in $\Re^3$. Consider the following two sequences of orientation vectors: $\{a_1, a_2, \ldots, a_p, v_1, v_2, \ldots, v_k, a_{p+1}, \ldots, a_n\}$ and $\{b_1, b_2, \ldots, b_q, v_1, v_2, \ldots, v_k, b_{q+1}, \ldots, b_n\}$. Define $f = \frac{k}{n+k}$, the fraction of the sequence corresponding to the set $\{v_i\}$. Then at the shift that brings the two copies of v_1 into alignment, the value of the expected orientation-RMS is at most $\sqrt{2(1-f)}$.*

Proof: We can provide an upper bound on the value of the orientation-RMS by computing the value of the RMS at a particular orientation: namely, the one which geometrically aligns the two copies of the sets $\{v_1, \ldots, v_k\}$. The crucial observation is that we can consider the vectors in $\{a_i\}$ to $\{b_j\}$ to have been generated after choosing this alignment; thus, the expected squared distance between each corresponding pair is 2, by Lemma 1. Thus, at this orientation, the expected sum of all the squared distances is $2(1-f)n$, and so the expected value of the RMS-distance is $\sqrt{2(1-f)}$. The value of the orientation-RMS, which minimizes over all possible orientations, is bounded by this quantity. $\square$

Consider two proteins of length 300 with a common substructure of length 45. This lemma implies that the expected orientation-RMS for these proteins should be $\leq \sqrt{1.7}$ or about 1.30. In fact, the orientation-RMS for proteins of length about 300 drops considerably below 1.30 even for smaller common substructures. Again, this is probably due to the nonrandom nature of real proteins.

Our strategy is thus to try all possible shifts of the two proteins looking for a drop in the orientation-RMS. Experiment indicates that this strategy is quite effective. For a given shift, we compute the correlation matrix and use this to compute the RMS-distance between the two sets of corresponding unit vectors. The FFT can be used to compute the matrix entries for each of the n shifts. It takes $O(n \log n)$ time to compute n matrices. For each of these matrices we can compute in constant time the RMS-distance between the corresponding unit vectors. Thus, in $O(n \log n)$ time we can compute the orientation-RMS for all possible shifts.

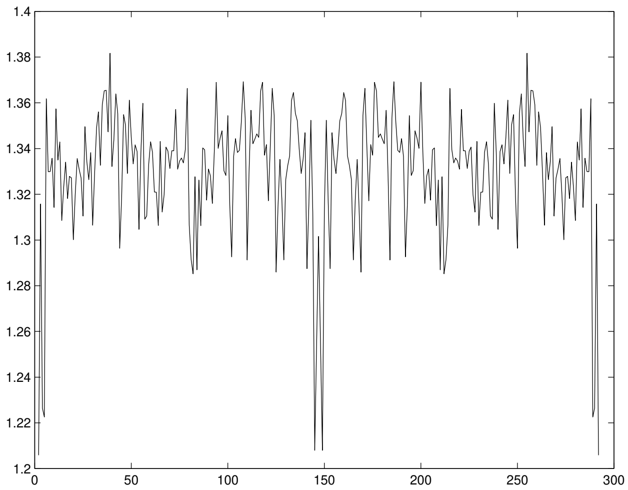

Figure 3: Orientation-RMS for each shift of 1rhd against itself.

The method so far can be used to detect which shifts might lead to a correspondence between common substructures, but we haven't described how that common substructure can be recognized. In other words, we have an efficient method for indicating when two proteins have a common substructure, but it doesn't actually tell us where the substructure is or how to rotate (in 3D) the structures so that the substructures have the same orientation. We do know the shift providing the correspondence between the two protein sequences, so we have the correct 1-to-1 map between the two proteins.

Consider the rotation needed to superimpose the substructures. If the substructures are sufficiently similar then this rotation is the same as that needed to superimpose subparts of the substructures. We can use this observation as a way to recognize the substructure. The idea is to compute the rotation matrices needed to bring each small piece of the first protein into rotational alignment with the corresponding piece of the second protein. For pieces that are part of the matching substructure, these rotation matrices will agree. We compute the rotation matrix at each unit vector by computing the rotation that most closely aligns the vector and its successor in the first protein with the corresponding vectors in the second protein.

To determine when two rotation matrices are similar, we use the Frobenius norm. The Frobenius norm of a matrix is the square root of the sum of the squares of all the matrix entries. This is relatively easy to compute and is invariant under rotation. Let Q and R be two rotation matrices (i.e., orthogonal matrices). Then the Frobenius norm of $Q - R$ is the same as the Frobenius norm of $QR^T - I$ where I is the identity matrix. This fits our intuition for how we want our matrix norm to behave: if Q and R are *close* as rotation matrices then QR^T should be *close* to the identity matrix.

For our problem, we are looking for sequences of matrices that agree. Thus it's sufficient to compare only adjacent matrices as we move down our protein sequence. This requires just $O(n)$ time.

3 Implementation and Experimental Results

We have implemented our algorithm in Matlab. The protein structures were taken from the Protein Data Bank (PDB) [1] and our results were compared against substructure similarities reported in

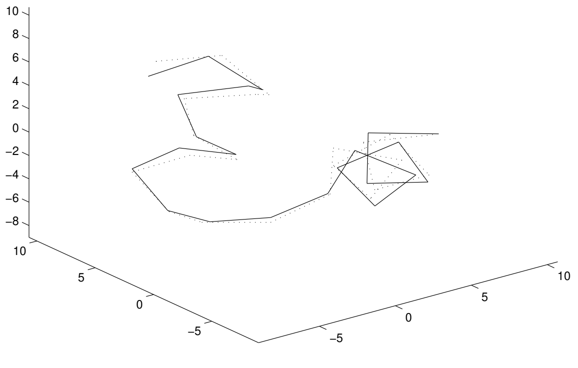

Figure 4: Subsequence match of 2cro(17–40) and 2wrp(62–85).

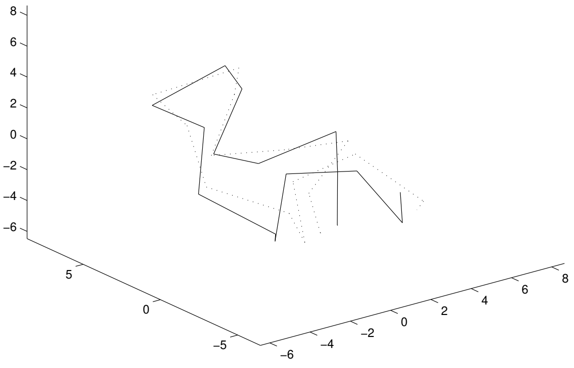

Figure 5: Subsequence match of 2cro(48–65) and 2wrp(65–82).

the literature. Some of our experiments are summarized in this section.

Assume the shorter protein is A and the longer one is B. The first residue in A is shifted to correspond in turn with each of the residues in B. At each shift, we truncate the unmatched portion of B. When protein A extends past the end of B, we wrap around to the beginning of B. We compute the rotation that minimizes the RMS-distance for the unit-vector representations of A and the truncated B.

Figures 1, 2, and 3 show the orientation-RMS for each possible shift of one protein against another. Our experiments have shown that only the first 10-20 lowest values are likely to represent shifts that bring significant substructures into correspondence. Our program takes just the best 20 shifts and proceeds to the next stage of finding the actual matching contiguous substructures for these shifts. To ensure that we have not ignored potential substructure matches, we have also run experiments to check for matching substructures for *all* the shifts, not just the best 20. So far, without exception, the other shifts did not yield any matching substructures of significant size.

Notice that in comparing 1rhd (293 residues) with itself the orientation-RMS plot (Figure 3) is symmetric about the median shift. This is caused by our cyclic shifting of residues and happens whenever we compare a protein with itself. Note that when comparing a protein with itself, we don't graph the zero-shift, since the orientation-RMS for that shift is 0. Including this shift would distort our scale.

In some structure comparisons (e.g., Figure 1: 2cro (65 residues) with 2wrp (104 residues)), a large consecutive substructure (22 out of 65 residues) is responsible for the drop in the orientation-RMS. In other cases (e.g., Figure 2: 1hsc (382 residues) with 2yhx (457 residues)), a number of small matching substructures (about three substructures of average length 12-13) were found for one shift and they cumulatively contributed to the drop in the orientation-RMS.

We compared our results with others in the literature. In Fischer et al. [4] a helix-turn-helix motif is found in tryptophan repressor (PDB code: 2wrp) and in phage 434 cro (PDB code: 2cro). Our graph showing the orientation-RMS for each shift is shown in Figure 1. We find a substrcture match between the same set of 23 consecutive residues (see Figure 4). Another long match our program discovers for these proteins is shown in Figure 5. Notice that these two matches involve the same section in 2wrp.

An internal duplication of motifs in the molecule of bovine liver rhodanese (PDB code: 1rhd, 293 residues) is well known ([15]). Our algorithm detects large matching substructures (see Figures 6 – 9 in the Appendix.) We discard substructures of length less than 12 so we don't report some of the matches found in [15]. (Substructures of length less than 12 are commonly portions of alpha-helices; these commonly occurring secondary structures are all of similar shape.)

In a third example, we compare the heat shock cognate protein (PDB code: 1hsc, 382 residues) with the protein hexokinase (PDB code: 2yhx, 457 residues). These proteins were compared in [5] though we find it hard to compare our results with theirs. A beta-sheet of 27 residues is found to match residues 193-220 in 1hsc with 63–100 in 2yhx (see Figure 10 in the Appendix).

4 Finding Common Protein Domains

We have preliminary results for composing larger structures (protein domains) by combining the smaller contiguous substructures detected using the above algorithm. Assume the substructure match program outputs s matching pairs of substructures. A large structure is a disjoint union of matching substructures that undergo the same transformation (rotation and translation) for the best RMS. Currently the match is found in a primitive way by comparing the $O(s^2)$ rotation matrices and finding a clique of similar matrices (according to the Frobenius norm) and then comparing the translation. (Notice that just comparing the rotations will yield bogus results on alpha-helix-rich structures, because a substructure a_1 of protein A might be similar to many substructures in B under the same rotation, but with different translations.) The matching substructures detected for 1rhd above were found to undergo similar transformation. We tested two SH2 protein fragments, the SH2 domain 1a1e (103 residues) and SH3-SH2 domain fragment of Human Bcr-Abl tyrosine Kinase Transferase (PDB code: 2abl, 164 residues). We found two large matching substructures (see Figures 11 and 12 in the Appendix) that sum up to 68 residues. Notice that the two matching substructure have a gap of 31 residues in 1a1e and 34 residues in 2abl.

Acknowledgements. The authors would like to thank Ron Elber, David Shalloway, and Steve Vavasis for very helpful discussions.

References

[1] F.C. Bernstein, T.F. Koetzle, G.J.B. Williams, E.F. Meyer, M.D. Brice, J.R. Rodgers, O. Kennard, T. Shimanouchi, M.Tasumi, *J. Mol. Bio.* 112(1977), pp. 535–542.

[2] N.P. Brown, C.A. Orengo, W.R. Taylor, "A protein structure comparison methodology," *Computer Chem.*, 1996.

[3] T.E. Creighton, *Proteins: Structures and Molecular Properties*, Freeman, 1992.

[4] D. Fischer, R. Nussinov and H. Wolfson, "3D substructure matching in protein molecules", *Proc. 3rd Intl. Symp. Combinatorial Pattern Matching*, Lecture Notes in Computer Science, 644, Springer-Verlag, pp. 459–477.

[5] H.M. Griendley, P.J. Artymiuk, D.W. Rice and P. Willett, "Identification of tertiary structure resemblance in proteins using a maximal common subgraph isomorphism algorithm", *J. Mol. Biol.* 229(1993), pp. 707 – 721.

[6] G.H. Golub and C.F. Van Loan, *Matrix Computations, 3rd Edition* (1996), Johns Hopkins University Press.

[7] L. Holm, C. Sander. “Mapping the protein universe.” Science 273 (1996), 595.

[8] L. Holm, C. Sander. “Searching protein structure databases has come of age.” *Proteins: Structure, Function, and Genetics*, 1994.

[9] Liisa Holm and Chris Sander, “Protein Structure Comparison by Alignment of Distance Matrices.” J. Mol. Biol. (1993) 233, 123-138.

[10] A.M. Lesk, *Acta Crystallographica*, A42(1986), pp. 110–113.

[11] Christine A. Orengo, Nigel P. Brown, and William R. Taylor, ”Fast Structure Alignment for Protein Databank Searching.” *PROTEINS: Structure, Function, and Genetics* 14:139-167 (1992).

[12] A. Sali, T. Blundell, *J. Mol. Bio.*, 212(1990), pp. 403–428.

[13] W.R. Taylor (editor), *Patterns in Protein Sequence and Structure*, Springer Series in Biophysics, Vol 7, Springer Verlag 1992.

[14] W.R. Taylor, C.A. Orengo, “Protein structure alignment,” *J. Mol. Bio.*, 208(1989), pp. 1–22.

[15] G. Vriend and C. Sander, “Detection of common three-dimensional substructures in proteins”, *PROTEINS: Structure, Function and Genetics*, 11(1991), pp. 52–58.

[16] D. Yee, K. Dill, “Families and the structural relatedness among globular proteins,” *Protein Science*, 2(1993).

5 Appendix

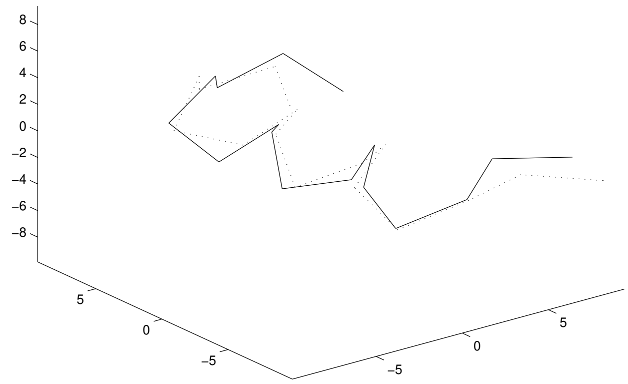

Figure 6: 1rhd internal match of residues 9–26 with 161–178.

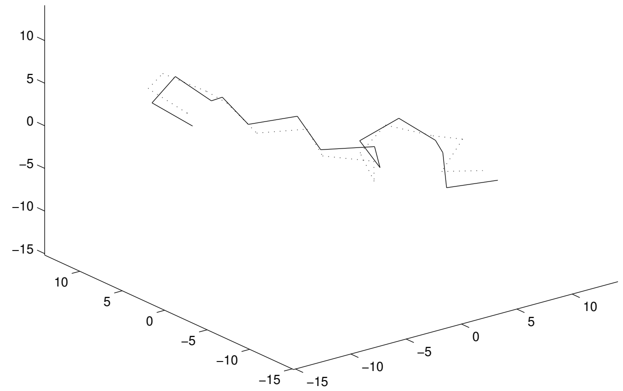

Figure 7: 1rhd internal match of residues 52–69 with 204–221.

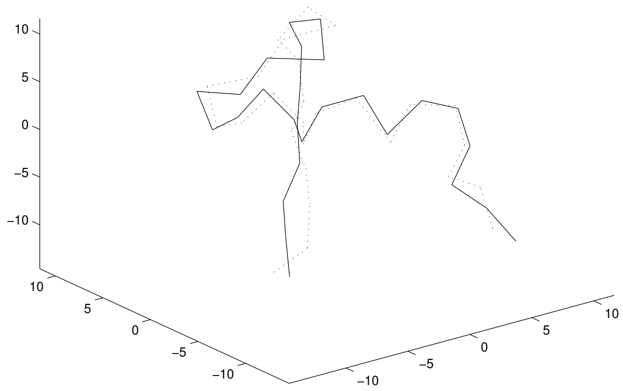

Figure 8: 1rhd internal match of residues 73–102 with 221–250.

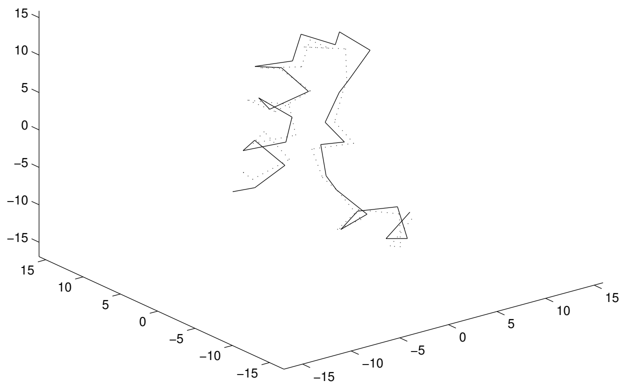

Figure 9: 1rhd internal match of residues 106–138 with 251–283.

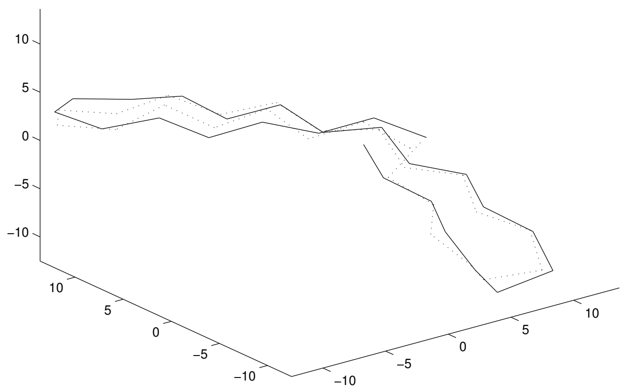

Figure 10: 1hsc(193–220) and 2yhx(63–90) superimposed.

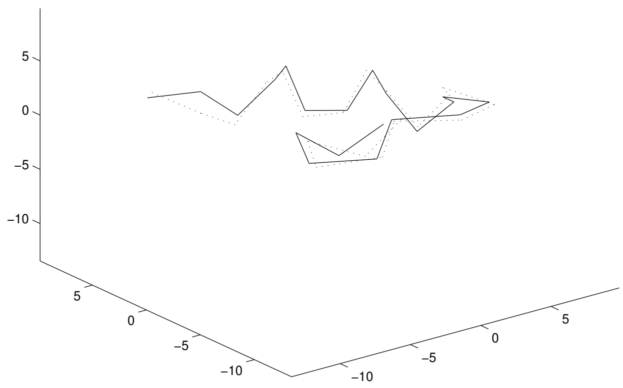

Figure 11: 1a1e(3–24) and 2abl(69–80) superimposed.

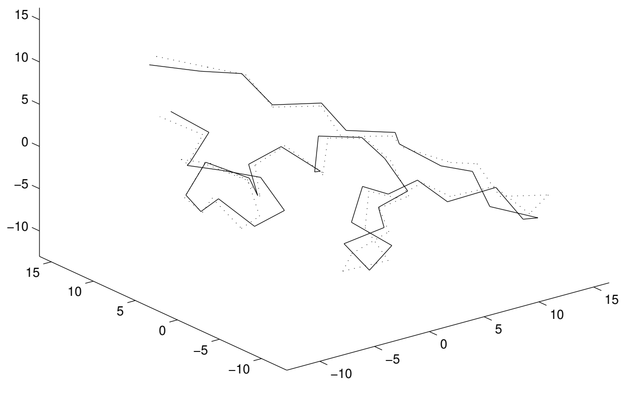

Figure 12: 1a1e(55–102) and 2abl(114–161) superimposed.