BIOINFORMATICS

Vol. 19 no. 5 2003, pages 625–634
DOI: 10.1093/bioinformatics/btg035

MINRMS: an efficient algorithm for determining protein structure similarity using root-mean-squared-distance

*Andrew I. Jewett, Conrad C. Huang and Thomas E. Ferrin**

Computer Graphics Laboratory, Department of Pharmaceutical Chemistry, University of California at San Fransisco, San Fransisco, CA 94143-0446 USA

Received on November 1, 2000; revised on January 30, 2001; October 17, 2002; accepted on November 7, 2002

ABSTRACT

Motivation: Existing algorithms for automated protein structure alignment generate contradictory results and are difficult to interpret. An algorithm which can provide a context for interpreting the alignment and uses a simple method to characterize protein structure similarity is needed.

Results: We describe a heuristic for limiting the search space for structure alignment comparisons between two proteins, and an algorithm for finding minimal root-mean-squared-distance (RMSD) alignments as a function of the number of matching residue pairs within this limited search space. Our alignment algorithm uses coordinates of alpha-carbon atoms to represent each amino acid residue and requires a total computation time of $O(m^3n^2)$, where m and n denote the lengths of the protein sequences. This makes our method fast enough for comparisons of moderate-size proteins (fewer than ∼800 residues) on current workstation-class computers and therefore addresses the need for a systematic analysis of multiple plausible shape similarities between two proteins using a widely accepted comparison metric.

Availability: See http://www.cgl.ucsf.edu/Research/minrms

Contact: tef@cgl.ucsf.edu

1 INTRODUCTION

The fact that amino acid sequence determines conformation in proteins with a high degree of redundancy has been observed for many years. Very different sequences can produce remarkably similar conformations. Recent inverse-protein-folding methods (Dahiya and Mayo, 1997) have produced artificial sequences that very closely reproduce the folded shape of a known protein, but have very low similarity with the known protein's sequence. Not surprisingly, mutations during the course of evolution alter the structure of a protein at a much slower rate than they disrupt the sequence content. Structural similarity between two proteins can reveal common ancestral heritage so distant that traditional comparisons based on sequence analysis alone fail to detect any relationship (Holm and Sander, 1993).

Over the last 25 years, a number algorithms have appeared that attempt to automate protein structural comparison and alignment using a wide variety of criteria. These algorithms usually fall into two categories:

Intermolecular (coordinate superposition): Algorithms that transform the atomic coordinates (i.e. relative global rotation and translation) of the two molecules so that their relevant parts are superimposed, and then compute the distance between the positions of the corresponding atoms in the two molecules. The algorithms of Falicov and Cohen (1996), Feng and Sippl (1996), Gerstein and Levitt (1998), Rao and Rossman (1973), Rossman and Argos (1976), Shindyalov and Bourne (1998), and Wu *et al.* (1998)† fall into this category. In general, intermolecular alignments do not handle molecular flexibility well.

Intramolecular: Algorithms that compare the relative positions or distances between atoms *within the same molecule* against corresponding relative positions or distances within the second molecule. Such algorithms do not depend on the relative orientation of the two molecules, but instead make use of '2D distance matrices' (Holm and Sander, 1993) 'local coordinate systems' (Taylor and Orengo, 1989) or 'curvature-matching' (Wu *et al.*, 1998).

*To whom correspondence should be addressed.

† The methods of Wu *et al.* (1998) and Shindyalov and Bourne (1998) are hybrids and make use of two different algorithms at different stages in the search, one intermolecular, the other intramolecular.

Cohen *et al.* (1980a) point out that it is not always possible to draw equivalent conclusions using these two approaches.

Although many of these approaches are computationally fast, they tend to produce alignments that are hard to interpret or explain (Gerstein and Levitt, 1998; Holm and Sander, 1993; Taylor and Orengo, 1989; Wu *et al.*, 1998). We have developed an algorithm that uses a straightforward, geometrically sensible, and widely accepted comparison metric for deciding when regions from two proteins are similar: the root-mean-squared-distance (RMSD) between aligned alpha-carbon atoms from the two proteins. Previous authors have expressed the desire to align structures using this metric (Falicov and Cohen, 1996; Wu *et al.*, 1998). Arguably, RMSD is the simplest intermolecular metric and is often quoted as a measure of alignment quality.

In contrast, many structural alignment algorithms described in the literature use *ad hoc* scoring criteria that, while certainly tending to have more favorable scores for alignments that match regions of similar shape, typically rely on scoring functions that use empirical parameters (e.g. 'gap penalties') whose geometrical significance is not rigorously defined or where the mathematical formulae used for scoring are not compelling. (There are exceptions; see Falicov and Cohen (1996).) For example, the choice of the number of pairwise amino acid equivalences may easily cause problems. With algorithms that use empirical parameters (Feng and Sippl, 1996; Gerstein and Levitt, 1998; Holm and Sander, 1993; Rao and Rossman, 1973; Shindyalov and Bourne, 1998; Taylor and Orengo, 1989; Wu *et al.*, 1998) the number of pairs of residues matched in the alignment is determined by the parameters chosen. An unfortunate choice of gap penalty could cause a very close alignment between two proteins to be discarded in favor of a poor alignment that matches more residues.

In our search for structure alignments, we limit ourselves to rigid-body superpositions, i.e. translations and rotations are applied to the entire structure rather than fragments. Hence our algorithm finds alignments that do not require internal alteration of the structures. Many algorithms forgo the rigid-body requirement. For example, Zuker and Somorjai (1989), describe an algorithm for aligning multiple fragments using multiple transformations. However, their algorithm does not address the problem that multiple fragments cannot be aligned completely independently. Taylor (1999) describes a double dynamic programming structure alignment algorithm which has the same shortcoming.

The methodology behind all of these algorithms would be irrelevant if the algorithms generated similar alignments. But for moderately distant structures this is not the case. Hen egg-white lysozyme (1LYZ) and T4 phage lysozyme (2LZM) provide a useful benchmark for protein structure comparison algorithms because the degree of structural similarity between these two proteins has been previously well noted (Matthews *et al.*, 1981; Rossman and Argos, 1976; Taylor and Orengo, 1989). Table 1 compares the similarity between the alignments generated by seven previously published algorithms. Alignment similarity is measured using sequence (%E) and structural criteria (3D). The %E metric represents the percentage of residue equivalences that are common to both alignments. The 3D metric represents the average displacement (in Angstroms) between the initial and final position of one of the structures after it has been optimally moved to superimpose the residue equivalences of the two structures. (See Section 3 for a precise definition.) The low percentage of residue equivalences and the high value of the 3D metric seen in Table 1 show substantial disagreement among the algorithms. In many cases the alignment produced by two different algorithms did not have a single pair of matched residues in common. This puts the significance of any one alignment in perspective, and underscores the need for a structure alignment method that is easy to interpret.

Our algorithm is characterized by explicit assumptions used in limiting the search space of the alignments we consider (see Section 2.1), so that it is known from the onset which solutions will or will not be considered. Unlike Taylor (1999), who uses stochastic methods to find a likely optimum, we have proven mathematically that our method finds solutions which are *minimal* in RMSD within the search space under consideration. Our MINRMS algorithm generates a *family* of alignments as a function of the number of residue equivalences without need of a gap penalty. As with some other algorithms (Holm and Sander, 1993; Feng and Sippl, 1996), we are also able to identify the second and third-best alignments. This additional information insures that interesting solutions will not be discarded.

We have also developed a visualization tool, AlignPlot (Huang *et al.*, 2000), to facilitate exploration of the potentially large number of alignments produced by MINRMS. Together, MINRMS and AlignPlot provide a facile way to explore the structure alignment space of even distantly related proteins and address concerns raised by others who have suggested it may not always be easy (or even possible) to find a single 'best' structural alignment (Feng and Sippl, 1996; Godzik, 1996; Orengo *et al.*, 1995). MINRMS has been used in the analysis of several protein superfamilies to discover possibly significant similarities between distantly related structures (Chiang *et al.*, 2003; Cantwell *et al.*, 2001; Babbitt, 2000).

Table 1. All-against-all comparison of results from other published algorithms of alignments between 1LYZ and 2LZM. The $\%E$ similarity metric is shown above the diagonal, while the 3D quality metric, in units of Angstroms, is shown below the diagonal

Algorithm	$Matt_{40}$	$Matt_{80}$	MINAREA	DALI	ALIGN	RA	TO	CE_{high}	CE_{medium}	CE_{low}
$Matt_{40}$	—	0%	0%	0%	0%	0%	0%	0%	0%	0%
$Matt_{80}$	28.11	—	0%	20%	0%	14%	16%	0%	20%	14%
MINAREA	18.14	21.08	—	0%	0%	0%	0%	0%	0%	0%
DALI	25.89	4.74	17.90	—	0%	22%	20%	0%	49%	0%
ALIGN	20.31	24.40	14.12	23.15	—	0%	0%	0%	0%	0%
RA	26.50	3.00	19.27	3.38	23.58	—	37%	0%	35%	3%
TO	25.30	4.42	18.34	2.61	23.33	3.32	—	0%	24%	11%
CE_{high}	19.71	30.00	17.55	24.8	21.14	26.04	23.69	—	0%	0%
CE_{medium}	27.51	1.38	20.43	4.12	24.10	3.12	3.66	28.79	—	31%
CE_{low}	26.15	4.01	19.03	4.48	23.74	3.02	3.32	26.76	4.13	—

2 SYSTEMS AND METHODS

Our MINRMS algorithm is based on inter-molecule structure alignment and requires the following two queries be addressed as prerequisites: (1) Given a fixed superposition, which residue pairs should be matched? And (2), how does one limit the number of potential superpositions to evaluate?

The first of these queries is difficult to answer if RMSD is the only comparison criterion, because matching one pair of residues will always result in an optimal value of RMSD (i.e. RMSD $= 0$). Thus, one cannot use RMSD to select N, the number of residues pairs to match. Rather than using some other auxiliary metric such as the Gerstein and Levitt probability value (Gerstein and Levitt, 1998) to select a single alignment per orientation, we chose to produce an alignment for each value of N. Users can then use their own judgment to decide on the relative importance of low RMSD and high number of matching residue pairs.

To address the second query, we appeal to domain-specific knowledge. When matching two molecular structures, a reasonable alignment will typically place some locally similar fragments from the two structures in close proximity; only unreasonable alignments will not succeed in matching any similar residue fragments from either molecule. Thus, we need only consider the set of superpositions that juxtaposes any pair of small fragments from the two molecular structures. By limiting our search to this subset of superpositions, we risk missing solutions which fail to closely match *any* fragments in favor of reducing global RMSD. But, by definition, these solutions cannot reveal any local structural similarity and therefore they are of limited use as structural alignments.

Basic Algorithm

Given these constraints, our algorithm is as follows:

(1) generate the set of initial superpositions to evaluate;

(2) for each candidate superposition of the two molecules, identify the minimal RMSD alignments;

(3) optimize the superposition of the molecules based on the best alignments obtained in step two.

We describe the details of each of these steps below.

2.1 Sampling Superposition Space

To generate the set of candidate superpositions, we superpose all fragments of four consecutive residues from one structure onto all fragments of four consecutive residues from the second structure. To superpose the two fragments, we use Diamond's method (Diamond, 1988) to align the alpha-carbon atoms from each of the four pairs of residues. This heuristic is similar to that used by Feng and Sippl (1996).

2.2 Identifying Matching Residue Pairs

To compute the residue equivalences, we use a dynamic programming algorithm similar to the algorithm by Needleman and Wunsch (1970). At this stage in the calculation, the two structures have been superimposed together and are not free to rotate. Let the position of the alpha carbons of these two structures (after superposition) be denoted by

$$\vec{r_1^A}, \vec{r_2^A}, \vec{r_3^A} \ldots \vec{r_m^A}$$
$$\vec{r_1^B}, \vec{r_2^B}, \vec{r_3^B} \ldots \vec{r_n^B}.$$

Given integers, i, j, and N, satisfying $1 \leq i \leq m$, $1 \leq j \leq n$, and $N \leq m$,, (where m, and n are the number of residues in each structure, and $m \leq n$), our objective is to choose two sequences of integers:

$$i_1, i_2, i_3, \ldots i_N, \text{ and}$$
$$j_1, j_2, j_3, \ldots j_N, \text{ where}$$
$$i_x < i_y \text{ iff } x < y, \text{ and } i_N \leq i, \text{ and}$$
$$j_x < j_y \text{ iff } x < y, \text{ and } j_N \leq j$$

which minimize the sum-squared-distance between matched residues

$$\sum_{x=1}^{N} \left| \vec{r}_{i_x}^{A} - \vec{r}_{j_x}^{B} \right|^2 .$$

Let $D_{N,i,j}$ denote the minimum possible sum-squared-distance between N pairs of matched residues, considering only the first i residues from structure A, and the first j residues from structure B. The minimum RMSD of any alignment containing N pairs of matched residues at this superposition is then $\sqrt{D_{N,m,n}/N}$.

2.2.1 Recursion For convenience, let $r_{i,j}$ denote the distance between the ith Cα from S_1 and the jth Cα from S_2 (at this particular orientation):

$$r_{i,j} \equiv \left| \vec{r}_i^{A} - \vec{r}_j^{B} \right| \tag{1}$$

$D_{N,i,j}$ obeys the recursive formula

$$D_{N,i,j} = \min \begin{cases} D_{(N-1),(i-1),(j-1)} + |r_{i,j}|^2 \\ D_{N,(i-1),j} \\ D_{N,i,(j-1)}. \end{cases} \tag{2}$$

The three cases in Equation 2 arise from the ways that the pair of residues i and j can be used in the alignment:

(1) residues i and j match;

(2) residue i does not participate in the alignment and does not add to the sum-squared-distance D;

(3) residue j does not participate in the alignment.

These three cases are not mutually exclusive. It's possible that neither residues i and j will participate in the alignment, but this is just a specific instance of case #2.

The base cases used to initiate the algorithm are

(1) $D_{0,i,j} = 0, \quad \forall i, j$

(2) $D_{N,i,j} = +\infty$ whenever $i < N$ or $j < N$.

Base case #1 is straightforward. The cumulative sum-squared-distance between zero pairs of residues is zero. Case #2 is a consequence of the fact that one can never have more than N correspondences in a set containing only N residues.

In contrast, the Needleman & Wunsch algorithm applies to sequence alignment and uses gap penalties to minimize arbitrary creation of alignment gaps because evolutionarily related sequences typically do not have many short insertions or deletions. Their dynamic programming algorithm loops over two variables, one for each sequence, and fills in a score matrix. The optimal alignment is recovered by back-tracing from the maximum value in the matrix.

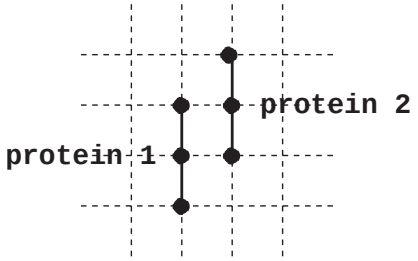

Fig. 1. Two schematic representations of protein structures of three residues each, with their Cα position fixed on a 1 Angstrom grid.

For structure alignment, we use a gap penalty of zero since we do not impose any requirement that matched residue pairs be contiguous. Where Needleman & Wunsch loops over two variables (sequence A, sequence B), our algorithm loops over three (structure A, structure B, number of matched residue pairs), and fills in a score pyramid. The score pyramid is actually many score matrices stacked on top of one another. The lowest layer represents the score matrix for matching only one pair of residues; each layer above represents the score matrix for matching an additional residue pair and is computed using the data from the layer below. If we regard each entry in the pyramid as a cell, then we can interpret Equation 2 above as stating that the value of a cell is derived from one of three adjacent cells. In our nomenclature, the adjacent cell whose value was used to compute the value of another cell is the 'predecessor' of the cell. Using the recursive property in Equation 2, every possible value of $D_{N,i,j}$ can be computed from previously calculated values. Afterwards, the optimal alignments are recovered by back-tracing the maximum value of each scoring matrix. Matches are made when the predecessor cell has a lower value of N.

2.2.2 Example Consider the table of $D_{N,i,j}$ values generated by aligning the two simple 'proteins' shown in schematic representation in Figure 1. Consider the calculation of the value $D_{2,3,3}$, midway through the table. This requires calculating the distance $|r_{3,3}|^2 = 2$, and choosing between

$D_{1,2,2} + |r_{3,3}|^2 = 3$,
$D_{2,3,2} = 2$, and
$D_{2,2,3} = 4$.

Note that by looping through the table in the right order, we've insured that $D_{1,2,2}$, $D_{2,3,2}$, and $D_{2,2,3}$ have already been calculated by the time we get to $D_{2,3,3}$. In this example, the predecessor of cell $D_{2,3,3}$ is $D_{2,3,2}$, which indicates that the optimal alignment with two matches ignores the third residue (i.e. $j = 3$) from the second protein.

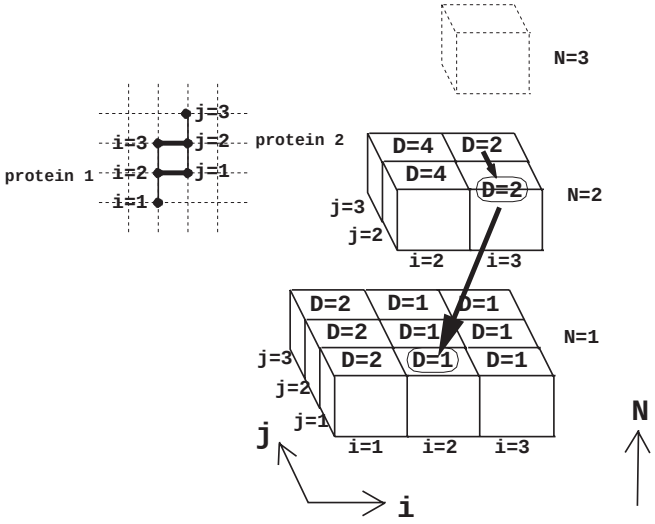

Fig. 2. Determining the lowest RMSD alignment with 2 residue equivalences. First, the table of all possible $D_{N,i,j}$ values is calculated using Equation 2. The alignment with 2 residue equivalences can then be found by examining the path through the table starting at position $D_{N=2,i=3,j=3}$. The predecessor of each cell in the path is pointed to with an arrow, and instances where an equivalence occurred are highlighted with ovals.

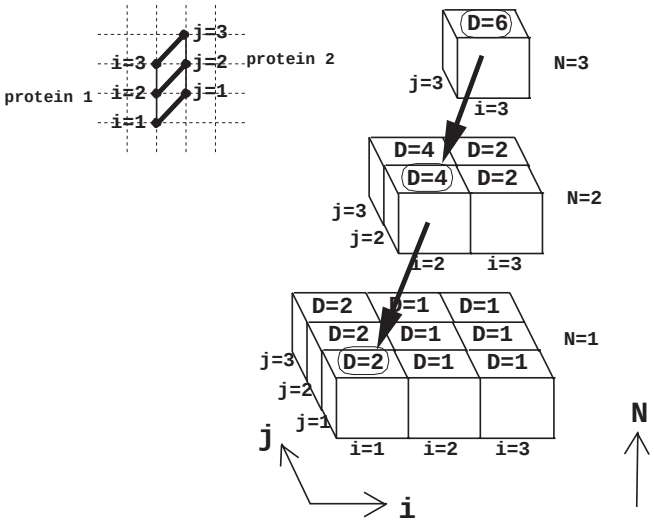

Fig. 3. Analogously, the lowest RMSD alignment with 3 residue equivalences can be found by looking at the path through the same table, this time starting at position $D_{N=3,i=3,j=3}$.

Figures 2 and 3 illustrate two different alignment paths for the proteins illustrated in Figure 1. Notice that increasing the number of matches from $N = 2$ to $N = 3$ results in a different (and more plausible) alignment. Adding a match is not always as trivial as just matching the next closest pair of available residues.

2.3 Refining superpositions

Candidate superpositions generated by matching small fragments are generated by using the optimal superposition matrix for the matching residue pairs in the fragment. Once a structure alignment is computed from the superposition, we have many more matching residue pairs. We can iteratively recompute a new superposition (by matching the residue pairs from the structure alignment) and recompute a new structure alignment using the new superposition, until some convergence criterion is met (e.g. RMSD of the new alignment is no better than the RMSD from the previous alignment). The generation of candidate superpositions may be considered a directed sampling of superposition space, while the iterative computation may be considered a refinement step for finding local minima.

2.4 Complexity analysis

Given two structures with n and m residues respectively, the cost of evaluating a single superposition is proportional to the cost of filling in the scoring pyramid, whose dimensions are n by m by the maximum number of matched residue pairs, which is the lesser of n and m, which, without loss of generality, we designate as m. Thus, the superposition evaluation cost is $O(m^2n)$. The number of superpositions that need to be evaluated is $(n - 4) \times (m - 4)$, so that the total complexity of our algorithm is $O(m^3n^2)$.

2.5 Reducing run time

The methods described in the previous section search a limited space of structure alignments. Unfortunately, the search space can still be sufficiently large that computation time is substantial. For example, for MINRMS to find the structure alignments between two proteins of length 468 and 380 requires 80 hours on a 250 MHz R10000 processor Silicon Graphics workstation. This computation time can be reduced to 40 minutes if we restrict our search to particular types of solutions[‡]. We describe three different methods of reducing the run time of our algorithm which, taken together, often result in a reduction in computation time from days to less than an hour.

2.5.1 Structure alignments between proteins The search-space-reducing heuristic described in Section 2.1 is applicable for aligning all types of molecules. For proteins, we can reduce the search space further by taking advantage of protein tertiary structure. When reasonable structure alignments exist between two proteins, they typically include all or part of the structural motifs of

[‡] The increase in speed cited in this example was made possible by using the same optimizations we use in Section 3.1.

the molecules. Since structural motifs have characteristic secondary structure elements (helices and beta sheets), we can further limit our heuristic to use only fragments of similar secondary structure type (helices to helices and sheets to sheets), rather than all fragments. In practice, computation time is reduced between 15- and 40-fold. The minor drawback of this approach is that structure alignments with large resulting RMSD (greater than 6 Angstroms), or with a low number of matched residue pairs (less than 20), may differ from alignments generated when the full search is used. In practice this has not proved to be a problem.

2.5.2 Restricting the number of matching residue pairs The dynamic programming algorithm described in Section 2.2 identifies minimal RMSD solutions for any number of matching residue pairs. However, structure alignments with a low number of matched residue pairs typically do not reflect good global structure alignment, because small local regions may be superposed without regard for the overall alignment. Structure alignments with very high number of matched residue pairs are also suspect because the residue alignments are driven more by the pair-count requirement than by spatial similarity. Thus, we can reasonably exclude these alignments from consideration. In practice, we allow a lower and/or an upper bound to be optionally specified for the number of matched residue pairs, and the effect on the algorithm is to compute only that part of the score pyramid that might possibly yield a solution satisfying the constraints. (The details are discussed at http://www.cgl.ucsf.edu/Research/minrms/N_MinMax.) The tradeoff with this approach is that one risks missing reasonable alignments at the edge of the search space. However visual inspection of alignment results using AlignPlot (Huang *et al.*, 2000) provides a good indication when too many solutions have been discarded.

In practice, this approach does not reduce the computation time or memory usage significantly unless either the lower bound is very high, or the upper bound is very low. However, this approach can be useful when it is known in advance that nearly all of the residues in at least one of the two structures are conserved (e.g. when searching for active site residues).

2.5.3 Imposing residue matching criterion The method described in Section 2 applies our dynamic programming algorithm to all candidate superpositions generated using the search-space-reducing heuristic from Section 2.1. All minimal RMSD alignments are computed, regardless of the alignment quality, even those that contain matched residue pairs which are spatially distant. However, the reliability of an alignment that contains a pair of matched residues as far apart, say, as 8 Angstroms is questionable. By providing an upper limit to the distance between matching residues, we can apply a Needleman & Wunsch-style filter to determine the maximum number of residue pairs that may be matched. (For details, see http://www.cgl.ucsf.edu/Research/minrms/nw.) These computed maxima may then be used as upper limits for the optimization described in Section 2.5.2. More importantly, if a lower limit is specified for optimization 2.5.2 then the computed maxima may be used to discard candidate superpositions that cannot possibly yield a solution with the required number of matching residue pairs.

The filter computation time for a candidate superposition runs in $O(mn)$ time, compared to $O(m^2n)$ time for the dynamic programming algorithm of Section 2.2. Thus, inclusion of the filter reduces computation time only if the overall time savings in discarding candidate superpositions is greater than the filter computation time itself. In practice, specifying a lower bound for optimization 2.5.2 typically is sufficient to make filtering worthwhile. For example, when aligning chain A of 2GLS against chain A of 1CRK (468 and 380 residues respectively), requiring that at least 100 pairs of residues must be within 8 Angstroms of each other eliminates 96.3% of all candidate superpositions. Even with the extra time needed for filtering, the net effect is a 16-fold speedup.

3 RESULTS

In the sections below, we examine three test systems and report and compare results from MINRMS and several other previously published algorithms. In these comparisons, we distinguish between *alignment quality*, which is the measure of 'reasonableness' or 'goodness' of a single alignment, and *alignment similarity*, which assesses the common features of two alignments. To evaluate alignment quality, we employ two numeric measures: RMSD and the probability score (P_{str}) from Levitt and Gerstein (Levitt and Gerstein, 1998). To evaluate alignment similarity, we define the metrics shown below.

Since structure alignments define both residue equivalences and 3D superposition, we can define comparison metrics for both measures. Given two structures, S_1 and S_2, and structure alignment A, we define three quantities:

$E(A)$ = The set of residue equivalences defined by A.

$R(S_i, A)$ = The set of residues from structure S_1 or S_2 used in A.

$\vec{T}(A, r)$ = The Cartesian coordinates of residue r in S_2. With structure S_1 fixed, this is the position of the Cα atom from residue r from S_2 after optimal superposition of S_2 onto S_1, as defined by A.

Using these definitions, we define the following quantities for comparing the structure alignments A_1 and A_2:

$$R_{both}(S_i, A_1, A_2) = \text{The set of residues from structure } S_1 \text{ or } S_2 \text{ matched in both } A_1 \text{ and } A_2.$$
$$= R(S_i, A_1) \bigcap R(S_i, A_2).$$
$$R_{either}(S_i, A_1, A_2) = \text{The set of residues from structure } S_1 \text{ or } S_2 \text{ matched in either } A_1 \text{ or } A_2.$$
$$= R(S_i, A_1) \bigcup R(S_i, A_2).$$

Using these quantities, we then define three measures of alignment similarity:

$$\%E(A_1, A_2) = \text{The lesser of the percentage of matched-residue-pairs from } A_1 \text{ that were also matched in } A_2, \text{ and the percentage of matched-residue-pairs from } A_2 \text{ that were also matched in } A_1.$$
$$= 100 \times \frac{|E(A_1) \bigcap E(A_2)|}{\max\{|E(A_1)|, |E(A_2)|\}}.$$

$$\%R(A_1, A_2) = \text{The lesser of the percentage of residues from } A_1 \text{ that were also matched in } A_2, \text{ and the percentage of residues from } A_2 \text{ that were also matched in } A_1.$$
$$= 100 \times \frac{|R_{both}(S_1, A_1, A_2)| + |R_{both}(S_2, A_1, A_2)|}{2 \times \max\{|E(A_1)|, |E(A_2)|\}}.$$

$$3D(A_1, A_2) = \text{The RMSD in position between the relative position of structure } S_2 \text{ when superimposed according to alignments } A_1 \text{ and } A_2. \text{ (Only the position of the C}\alpha \text{ atoms belonging to residues matched in either alignment are considered.)}$$
$$= \sqrt{\sum_{r \in R_{either}(S_2, A_1, A_2)} \frac{|\vec{T}(A_1, r) - \vec{T}(A_2, r)|^2}{|R_{either}(S_2, A_1, A_2)|}}.$$

$\%E$ and $\%R$ quantify the alignment similarity in sequence space, while 3D does so in Cartesian space. $\%E$ measures whether the two alignments identify similar matched residue pairs, while $\%R$ measures whether they identify the same residues as being important for the alignment. 3D quantifies whether the alignments position the important residues in a similar way. Both types of measures are necessary because it is possible for two alignments to match different residues and still have similar Cartesian superposition.

3.1 Test systems

We compare MINRMS against seven other algorithms (Table 2) using three pairs of structures (Table 3). To evaluate the results from MINRMS against another algorithm for a pair of structures, we first obtain the alignment from the reference algorithm and then compute the RMSD between matched residue pairs and the Levitt and Gerstein P_{str} probability score. We then run MINRMS to obtain our family of alignments. Our MINRMS runs were restricted to superpositions between fragments with matching secondary structure (optimization 2.5.1) and require that the alignments match at least 30% of the residues from the smaller structure (optimization 2.5.2), and that matched residues are no farther than 8A apart (optimization 2.5.3). MINRMS execution times are shown in Table 3.

To compare the alignment quality, RMSD from the MINRMS alignment is reported (using the same number of residue equivalences as the reference algorithm), as is the best P_{str} probability score of all MINRMS alignments. To evaluate alignment similarity, we compute $\%E$, $\%R$, and 3D between the reference algorithm and all MINRMS alignments. The best value for each similarity measure is reported as well as the number of equivalences made in the corresponding MINRMS alignment.

3.2 Lysozyme

The wide disagreement between algorithms evident in Table 1 is reflected in the wide range of RMSD and P_{str} values shown in Table 4. Table 1 shows that in the lysozyme test case, the alignment generated by CE_{high} was completely dissimilar to the alignments from all other algorithms ($\%E = 0$, 3D $\geq$ 17.5A). The same can be said for the alignments from each of ALIGN, MINAREA, and $Matthews_{40}$. The remaining algorithms share some similarity ($\%E$ = 14–49%, 3D = 1.38–4.71A). As seen in Table 4, MINRMS does not produce any alignment that is similar to the alignments of ALIGN, CE_{high} or MINAREA. The MINRMS alignment with 119 matched pairs of residues includes the 40 residue equivalences matched in the alignment from $Matthews_{40}$ as a subset, but superimposes the structures very differently (3D $\geq$ 15A). MINRMS does find alignments similar to the other algorithms ($\%E$ = 34–45%, 3D = 1.45–3.54A). These results show that in a single run, MINRMS produces a range of plausible alignments consistent with multiple reference algorithms.

3.3 Translational symmetry

The cytokine proteins 1RMI and 1LKI are much more structurally similar than the lysozyme proteins discussed above. They represent one of the 100 pairs of structures from the FSSP database (as of 1994) with the lowest FC ratio (Falicov and Cohen, 1996), and less than

Table 2. Previously published structure alignment algorithms used for alignment comparisons

Algorithm	inter-or-intra	Comments
$Matthews_{40}$	inter	The lowest-RMSD match between fragments of 40 consecutive residues from each protein (Matthews *et al.*, 1981).
$Matthews_{80}$	inter	The lowest-RMSD match between fragments of 80 consecutive residues from each protein (Matthews *et al.*, 1981).
MINAREA	inter	The MINAREA program generates a surface between two chains (Falicov and Cohen, 1996), as well as two separate alignments. The alignments used in this paper are the ones with lower RMSD.
DALI	intra	Generated using the DALI server available on the web, which uses an algorithm described in Holm and Sander (1993).
ALIGN	inter	Generated by the ALIGN server available on the web and described in Gerstein and Levitt (1998).
RA	inter	Published in Rossman and Argos (1976).
TO	intra	Published in Taylor and Orengo (1989).
CE_{high}	inter	Generated by the CE server (Shindyalov and Bourne, 1998) using 'high similarity' settings.
CE_{medium}	inter	Generated by the CE server using 'medium similarity' settings.
CE_{low}	inter	Generated by the CE server using 'low similarity' settings.

Table 3. Proteins used for alignment comparisions, their relative sequence identity, and the time required by MINRMS to align the structures

Proteins	Lengths	% Sequence Identity	MINRMS Run Time	Comments
2LZM 1LYZ	164 129	18%	1.2 min	Well studied system.
1LKI 1RMI	172 160	16.0%	6.9 min	Two multi-helix-bundles exhibiting translational symmetry.
1CNV 1XAS	283 295	14.9%	14.3 min	Two TIM-barrels exhibiting rotational symmetry.

Table 4. Comparison of alignments between hen egg-white lysozyme (1LYZ) and T4 phage lysozyme (2LZM)

Algorithm Results				MINRMS Results		Similarity					
Algorithm	N	RMSD	$\log_{10}(P_{str})$	RMSD(N)	Best $\log_{10}(P_{str})$	Highest %E	(N_E)	Highest %R	(N_R)	Lowest 3-D	(N_{3D})
$Matthews_{40}$	40	3.8	−3.34	1.3		34%	(119)	35%	(113)	15.07	(126)
$Matthews_{80}$	80	6.1	−3.64	3.2		38%	(101)	76%	(85)	3.87	(46)
MINAREA	44	4.6	−0.21	1.4		2%	(127)	35%	(113)	4.06	(127)
DALI	70	3.4	−5.11	2.6		45%	(95)	83%	(70)	2.18	(95)
ALIGN	57	3.7	−4.15	1.9	−5.45	0%		48%	(58)	8.64	(114)
RA	78	4.2	−4.73	3.1	($N = 82$)	42%	(94)	83%	(78)	1.75	(88)
TO	88	6.8	−2.34	3.8		41%	(94)	79%	(89)	2.08	(101)
CE_{high}	48	4.2	−3.04	1.6		0%		39%	(113)	18.94	(119)
CE_{medium}	80	4.7	−5.07	3.2		34%	(95)	79%	(85)	3.54	(101)
CE_{low}	96	5.8	−4.40	4.7		43%	(97)	90%	(98)	1.45	(105)

25% sequence identity. Both are bundles of parallel helices, with 1RMI being slightly longer than 1LKI. This translational symmetry creates difficulties when deciding how to best superimpose the two structures. Table 5 shows the results of this test system.

Alignments from the various reference algorithms differ significantly, often as a result of sliding one structure along the other by one or two turns of a helix. Of these possible results, at least two groups of plausible and qualitatively different solutions are found in the 159 alignments generated by MINRMS.

Alignments containing less than 122 equivalences superimposed the turns on one end of the two multi-helix bundles. These solutions were similar to two of the solutions produced by the CE server, using the 'high' and 'low' similarity settings. Alignments containing between

Table 5. Comparison of alignments of two cytokines: leukemia inhibitory factor (1LKI) and interferon-beta (1RMI)

Algorithm Results				MINRMS Results		Similarity Highest		Highest		Lowest	
Algorithm	N	RMSD	$\log_{10}(P_{str})$	RMSD(N)	Best $\log_{10}(P_{str})$	$\%E$	(N_E)	$\%R$	(N_R)	3-D	(N_{3D})
MINAREA	109	2.5	−5.41	2.0		39%	(127)	79%	(127)	1.63	(141)
DALI	129	3.2	−9.64	2.6		28%	(152)	85%	(129)	4.45	(151)
ALIGN	143	3.3	−10.67	3.2	−10.60	69%	(141)	96%	(144)	0.85	(139)
CE_{high}	137	3.5	−10.19	2.9	$(N = 147)$	68%	(119)	91%	(138)	1.46	(121)
CE_{medium}	137	3.2	−10.24	2.9		85%	(143)	95%	(138)	0.51	(131)
CE_{low}	132	3.4	−9.81	2.7		70%	(119)	89%	(135)	1.76	(118)

Table 6. Comparison of alignments between concanavalin B seed protein (1CNV) and xylanase A (1XAS)

Algorithm Results				MINRMS Results		Similarity Highest		Highest		Lowest	
Algorithm	N	RMSD	$\log_{10}(P_{str})$	RMSD(N)	Best $\log_{10}(P_{str})$	$\%E$	(N_E)	$\%R$	(N_R)	3-D	(N_{3D})
MINAREA	169	3.2	−9.35	2.4		72%	(168)	87%	(170)	0.71	(133)
DALI	208	3.8	−14.48	3.2		59%	(206)	91%	(209)	0.94	(110)
ALIGN	231	4.4	−15.83	4.0	−15.30	60%	(223)	94%	(231)	0.93	(163)
CE_{high}	204	4.1	−13.33	3.1	$(N = 238)$	38%	(254)	84%	(204)	2.19	(241)
CE_{medium}	212	4.7	−13.49	3.3		0%		77%	(212)	10.06	(109)
CE_{low}	236	5.0	−14.10	4.2		45%	(251)	92%	(237)	2.82	(241)

122 and 148 equivalences used a different superposition which brought the turns on the other end of the helix bundles into close proximity. This was in agreement with the solutions generated by ALIGN and MINAREA. A third solution, which was unique to DALI, produced an alignment that was displaced by one helical turn from the second solution.

3.4 Rotational symmetry

We encountered similar results results when aligning seed protein (1CNV) with xylanase A (1XAS) (Table 6). These two proteins are TIM-barrels with eight-fold rotational symmetry and low sequence similarity (14.9% sequence identity). For this discussion, we denote the helices in these TIM-barrel proteins as A through H sequentially around the barrel. The resulting alignments generated by MINRMS with less than 110 matched residues superimposed the helical barrels with a 90 degree rotation about the symmetry axis, matching helix A from 1CNV with helix C from 1XAS. Solutions with more than 110 matched residues superimposed the helical barrels with no rotation. The alignments of DALI, CE_{low}, CE_{high}, MINAREA, and ALIGN aligned the proteins in this same way. CE_{medium} found the unique solution matching the TIM barrels with a 45 degree rotation about the symmetry axis, matching helix A from 1CNV with helix B from 1XAS.

Of the eight plausible alignments obtainable from rotational symmetry, MINRMS reports only two. Because all pairs of fragments were matched together, (see Section 2.1), MINRMS did consider each of the other six superpositions; however, alignments made at those other superpositions had less favorable RMSD.

4 DISCUSSION

Tables 4–6 show that MINRMS finds alignments with lower RMSD for the same number of residue equivalences than the reference algorithms. These results are to be expected because MINRMS is designed to minimize RMSD while the other algorithms are not. However, MINRMS also finds alignments whose P_{str} values are among the best. The only algorithm to find alignments with better P_{str} values is ALIGN, which was developed to optimize the P_{str} metric. In fact, it is surprising to note that for the lysozyme test case, MINRMS found an alignment with a lower P_{str} than ALIGN; this interesting result is probably due to the more extensive orientation sampling of MINRMS.

In general, MINRMS finds a similar solution to all alignments having low P_{str}. The two main exceptions are: an alignment generated by ALIGN which scores well by its own metric (Table 4) and an alignment generated by CE_{medium} (Table 6), both of which are dissimilar to the alignments from every other reference algorithm.

The last two test cases were pairs of proteins that were difficult to align because of their symmetry (translational or rotational). In both cases MINRMS found multiple, significantly different solutions, informing the user that there wasn't a single unique answer. This extra information puts a perspective on the significance of any one alignment. Reporting a single alignment between such proteins could be misleading. Of course, there is no guarantee that all of the possible interesting solutions will be among the lowest RMSD alignments reported by MINRMS. However, because of the extensive way we sample orientation space (see Section 2.1), we can guarantee that all of the relevant alignments are considered in the search.[§]

MINRMS is not the only algorithm which can generate multiple solutions. Both the DALI algorithm (Holm and Sander, 1993) and Feng & Sipple's algorithm (Feng and Sippl, 1996) can save several top scoring alignments. Nearly all structural alignment algorithms contain parameters that can be adjusted to produce more than one solution. The strength of MINRMS is that it systematically finds optimal alignments within a large but bounded search space using a metric that is easy to understand.

ACKNOWLEDGEMENTS

This research was supported by DOE DE-FG03-96ER62269 and NIH P41-RR01081 (T.E. Ferrin, P.I.). We thank P. Babbitt, C. Oshiro, N. Ulyanov, and S. Mooney for their many helpful suggestions. We also thank A. Falicov for his helpful e-mail discussions, F. Cohen for access to the MINAREA software, L. Holm and C. Sander for access to, and help using, the DALI server, and W. Krebs, M. Gerstein and M. Levitt, for access to and help with their ALIGN server.

REFERENCES

Babbitt,P.C. (2000) Reeingineering the glutathione S-transferase scaffold: A rational design strategy pays off, (Commentary). *Proc. Natl Acad. Sci. USA*, **97**, 10298–10300.

Cantwell,J.S., Novak,W.R., Wang,P-F., McLeish,M.J., Kenyon,G.L. and Babbitt,P.C. (2001) Mutagenesis of two acidic active site residues in human muscle creatine kinase: implications for the catalytic mechanism. *Biochem.*, **40**, 3056–3061.

Chiang,R.A., Meng,E.C., Huang,C.C., Ferrin,T.E. and Babbitt,P.C. (2003) The structure-superposition database. *Nucleic Acids Res.*, **31**, 505–510.

Cohen,F.E. and Sternberg,M.J.E. (1980a) On the prediction of protein structure: the significance of the root mean squared deviation. *J. Mol. Biol*, **138**, 321–333.

Dahiya,B. and Mayo,S. (1997) De Novo protein design: fully automated sequence selection. *Science*, **278**, 82–87.

Diamond,R. (1988) A note on the rotational superposition problem. *Acta Cryst.*, **44**, 211–216.

Falicov,A. and Cohen,F.E. (1996) A surface of minimum area metric for the structural comparison of proteins. *J. Mol. Biol*, **258**, 871–892.

Feng,Z.K. and Sippl,M.J. (1996) Optimal superimposition of protein structures: ambiguities and implications. *Fold. Design*, **1**, 123–132.

Gerstein,M. and Levitt,M. (1998) Comprehensive assessment of automatic structural alignment against a manual standard, the scop classification of proteins. *Protein Sci.*, **7**, 445–456.

Godzik,A. (1996) The structural alignment between two proteins: is there a unique answer? *Protein Sci.*, **5**, 1325–1338.

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

Huang,C.C., Novak,W.J., Babbitt,P.C., Jewett,A.I., Ferrin,T.E. and Klein,T.E. (2000) Integrated tools for structural and sequence alignment and analysis. *Pac. Symp. Biocomput.*, **5**, 227–238. See http://www.cgl.ucsf.edu/chimera for additional information.

Levitt,M. and Gerstein,M. (1998) A unified statistical framework for sequence comparison and structure comparison. *Proc. Natl Acad. Sci. USA*, **95**, 5913–5920.

Matthews,B.W., Remington,S.J., Grutter,M.G. and Anderson,W.F. (1981) Relation between hen egg white lysozyme and bacteriophage T4 lysozyme: evolutionary implications. *J. Mol. Biol*, **147**, 545–558.

Needleman,S.B. and Wunsch,C.D. (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J. Mol. Biol*, **48**, 443–453.

Orengo,C.A., Swindells,M.B., Michie,A.D., Zvelebil,M.J., Driscoll,P.C., Waterfield,M.D. and Thornton,J.M. (1995) Structural similarity between the pleckstrin homology domain and verotoxin; The problem of measuring and evaluating structural similarity. *Protein Sci.*, **4**, 1977–1983.

Rao,S.T. and Rossman,M.G. (1973) Comparison of super-secondary structures in proteins. *J. Mol. Biol*, **76**, 241–256.

Rossman,M.G. and Argos,P. (1976) Exploring Structural Homology of Proteins. *J. Mol. Biol*, **105**, 85–91.

Sellers,P. (1974b) Theory and computation of evolutionary distances. *SIAM J. Appl. Math*, **26**, 787.

Shindyalov,I.N. and Bourne,P.E. (1998) Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. *Protein Eng.*, **11**, 739–747.

Taylor,W.R. (1999) Protein structure alignment using iterated double dynamic programming. *Protein Sci.*, **8**, 654–665.

Taylor,W.R. and Orengo,C.A. (1989) Protein structure alignment. *J. Mol. Biol*, **208**, 1–22.

Wu,T.D., Schidler,S.C., Hastie,T. and Brutlag,D.L. (1998) Modeling and superposition of multiple protein structures using affine transformations: analysis of the globins. *Pac. Symp. Biocomput.*, **3**, 509–522.

Zuker,M. and Somorjai,R.L. (1989) The alignment of protein structure in three dimensions. *Bull. Math. Biol.*, **51**, 55–78.

[§] Instead of discarding these alignments, MINRMS can easily be modified to retain them and produce an even larger set of alignments for the user to consider. However, in this paper we consider only the alignments of lowest RMSD.