Protein Science (1998), 7:445–456. Cambridge University Press. Printed in the USA.

Comprehensive assessment of automatic structural alignment against a manual standard, the scop classification of proteins

MARK GERSTEIN[1] AND MICHAEL LEVITT[2]

[1]Molecular Biophysics & Biochemistry Department, P.O. Box 208114, Yale University, New Haven, Connecticut 06520-8114
[2]Structural Biology Department, Stanford University, Stanford, California 94305

(RECEIVED August 6, 1997; ACCEPTED October 16, 1997)

Abstract

We apply a simple method for aligning protein sequences on the basis of a 3D structure, on a large scale, to the proteins in the scop classification of fold families. This allows us to assess, understand, and improve our automatic method against an objective, manually derived standard, a type of comprehensive evaluation that has not yet been possible for other structural alignment algorithms. Our basic approach directly matches the backbones of two structures, using repeated cycles of dynamic programming and least-squares fitting to determine an alignment minimizing coordinate difference. Because of simplicity, our method can be readily modified to take into account additional features of protein structure such as the orientation of side chains or the location-dependent cost of opening a gap. Our basic method, augmented by such modifications, can find reasonable alignments for all but 1.5% of the known structural similarities in scop, i.e., all but 32 of the 2,107 superfamily pairs. We discuss the specific protein structural features that make these 32 pairs so difficult to align and show how our procedure effectively partitions the relationships in scop into different categories, depending on what aspects of protein structure are involved (e.g., depending on whether or not consideration of side-chain orientation is necessary for proper alignment). We also show how our pairwise alignment procedure can be extended to generate a multiple alignment for a group of related structures. We have compared these alignments in detail with corresponding manual ones culled from the literature. We find good agreement (to within 95% for the core regions), and detailed comparison highlights how particular protein structural features (such as certain strands) are problematical to align, giving somewhat ambiguous results. With these improvements and systematic tests, our procedure should be useful for the development of scop and the future classification of protein folds. Supplementary material is available at http://bioinfo.mbb.yale.edu/align.

Keywords: bioinformatics; databank comparison; molecular evolution; protein fold; sequence alignment

Structural alignment consists of establishing equivalences between the residues in two different proteins, as is the case with conventional sequence alignment. However, this equivalence is determined principally on the basis of the three-dimensional coordinates corresponding to each residue, not on the basis of the amino acid "type" of the residue. The general idea of structural alignment has been around since the first comparisons of the structures of myoglobin and hemoglobin (Perutz et al., 1960). Systematic structural alignment began with the analysis of heme binding proteins and dehydrogenases by Rossmann and colleagues (Rossmann et al., 1975; Rossmann & Argos, 1975; Argos & Rossmann, 1979). Currently, there are two basic reasons for wanting to perform this operation.

First, the number of known structures is large and growing rapidly (>8,000 domains in the Protein Databank, expected to exceed 10,000 soon) (Orengo, 1994; Murzin et al., 1995; Bernstein et al., 1977; Holm & Sander, 1997). Both for understanding and for applications such as comparative modelling (Sanchez & Sali, 1997), it is advantageous to organize all the structures into fold families. A number of databases currently do this: FSSP and Entrez-MMDB cluster structures purely on the basis of automatic comparison programs (Holm & Sander, 1993a, 1994, 1996; Gibrat et al., 1996; Hogue et al., 1996; Schuler et al., 1996). Scop does the same thing manually, based on visual inspection of human experts (Murzin et al., 1995). And CATH and HOMALDB adopt an intermediate approach, using both automatic and manual methods (Overington et al., 1993; Orengo et al., 1994; Sali & Overington, 1994).

Second, structural alignment can be used as a "gold standard" for sequence alignment and threading. How does one know if a purely sequence-based alignment is correct? Or which parts of two proteins can be aligned? The current belief is that this is best done by consulting a structural alignment, particularly for alignments of highly diverged sequences (Vogt et al., 1995; Chothia & Gerstein, 1997). This second use of structural alignment tends to focus on

Reprint requests to: Mark Gerstein, Molecular Biophysics & Biochemistry Department, P.O. Box 208114, Yale University, New Haven, Connecticut 06520-8114; e-mail: Mark.Gerstein@yale.edu.

the accuracy of an alignment given that one already knows that two structures are similar.

Existing methods for structural alignment

Because of their obvious utility, a large number of different procedures for automatic structural alignment and comparison have been developed (Remington & Matthews, 1980; Satow et al., 1987; Artymiuk et al., 1989; Taylor & Orengo, 1989; Sali & Blundell, 1990; Vriend et al., 1991; Russell & Barton, 1992; Grindley et al., 1993; Holm & Sander, 1993a; Godzik & Skolnick, 1994; Feng & Sippl, 1996; Falicov & Cohen, 1996; Gibrat et al., 1996; Cohen, 1997).

To understand these procedures, it is useful to compare structural alignment with the much more thoroughly studied methods for sequence alignment (Doolittle, 1987; Gribskov & Devereux, 1992). Both sequence and structure alignment methods produce an alignment that can be described as an ordered set of equivalent pairs (i,j) associating residue i in protein A with residue j in protein B. Both methods allow gaps in these alignments that correspond to non-sequential i (or j) values in consecutive pairs—i.e., one has pairs like (10,20) and (11,22). And both methods reach an alignment by optimizing a function that scores well for good matches and badly for gaps. The major difference between the methods is that the optimization used for sequence alignment is globally convergent, whereas that used for structural alignment is not. This is the case for sequence alignment because the optimum match for one part of a sequence is *not* affected by the match for any other part. Structural alignment fails to converge globally because the possible matches for different segments are tightly linked as they are part of the same rigid 3D structure. For this reason, the alignment found by a structural alignment algorithm can depend on the initial equivalences, whereas in sequence alignment there is no such dependence.

The lack-of-convergence problem has led to a large number of different approaches to structural alignment, the methods differing in how they attack the problem. However, no current algorithm can find the globally optimum solution all the time; the convergence problem remains unsolved in the general case. The methods also differ in the function they optimize (the equivalent of the amino acid substitution matrix used in sequence alignment) and how they treat gaps.

Some of the methods effectively compare the respective distance matrices of each structure, trying to minimize the difference in *intra*-atomic distances for selected aligned substructures (Taylor & Orengo, 1989; Sali & Blundell, 1990; Holm & Sander, 1993a). In contrast, our method, which is derived from that of Cohen (Satow et al., 1987; Cohen, 1997), directly tries to minimize the *inter*-atomic distances between two structures. A similar approach is taken in minimizing the "soap-bubble area" between two structures (Falicov & Cohen, 1996). Other methods involve further techniques, such as geometric hashing or lattice fitting (Artymiuk et al., 1989; Godzik & Skolnick, 1994; Gibrat et al., 1996).

The importance of manual standards

How well do the current structural alignment programs perform? Although particular programs have uncovered many interesting similarities in individual cases (e.g., globin-colicin-A, Holm & Sander, 1993b; adenylyl cyclase-polymerase, Artymiuk et al., 1997; Bryant et al., 1997), it has not been possible to see how well the programs perform overall, in an aggregate, statistical fashion against a set of objective standards. This is because up to now suitable standards did not exist. However, the recently created scop classification of protein structures provides such a suitable standard (Murzin et al., 1995; Brenner et al., 1996; Hubbard et al., 1997). It consists of thousands of documented similarities between known protein structures based purely on visual inspection. Here, we endeavor to test our automatic method of structural comparison against the known similarities in scop. This provides, for the first time, a comprehensive sense of how a uniformly applied automatic procedure does against the manual standard. It also allows us to see what type of similarities are especially hard to detect and to optimize our procedure in a systematic fashion.

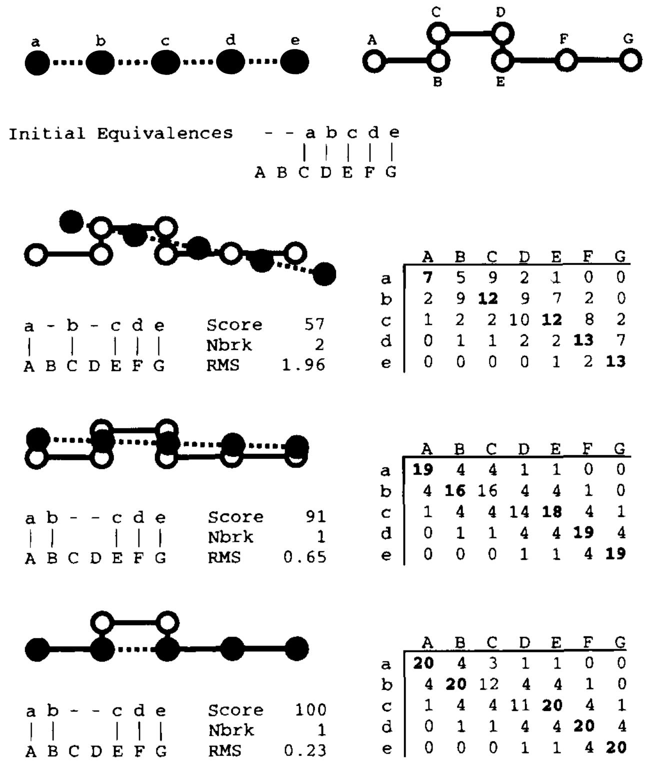
After a program has found a structural similarity, the next question one asks is how correct is the alignment. This is especially important if one wants to use results of structural alignment as a "gold standard" to evaluate a sequence-alignment or threading algorithm. It is surprisingly difficult to answer this question in detail because many parts of two similar proteins (e.g., loops) may not be alignable at all. Some recent results have highlighted the ambiguities in structural alignment and even suggested that unique alignments do not exist (Orengo et al., 1995; Feng & Sippl, 1996; Godzik, 1996). However, we take the perspective that unique alignments exist for the essential "core" regions of two similar proteins. As was the case with the detection of similarities, it is essential to compare automatic alignments against manual standards in an objective and systematic fashion. Here, we test a selection of the alignments derived from scop against corresponding manual alignments from the literature.

Results

Systematic elaboration of a simple procedure (search then iterate)

As shown in Figure 1, the basic procedure we use for structural alignment is very simple. It is very much like classic Needleman-Wunsch sequence alignment (Needleman & Wunsch, 1971). It consists of building a similarity matrix S_{ij} based on the interatomic distances between each atom i in the first structure and each atom j in the second. Then dynamic programming is applied to this matrix to find the optimal global alignment. If this were sequence alignment, we would be done, as the similarity matrix, which depends only on the two sequences, is constant. However, in structural alignment, the matrix depends on the relative 3D positioning of the two structures, which in turn, depends on how they have been previously aligned, so the procedure must be iterated until it converges. As we will describe below, this simple procedure is usually able to arrive at the correct alignment. However, there are exceptions. To handle these, we modified our basic procedure in two ways: through an expanded search and through using additional methods to build the similarity matrix. Because of the simplicity of the basic procedure these modifications can be rationalized directly in terms of features of protein structure.

Originally, our search consisted of starting at five reasonably chosen points, described in the methods. Here, we expand the search by allowing additional starting points and, in certain difficult cases, only aligning a section of the bigger of the two proteins. In the basic method, the similarity matrix depended only on the distance between alpha carbons (method "Cα"). Here, we elabo-



```
Initial Equivalences   - - a b c d e
                           | | | | |
                       A B C D E F G
```

```
a - b - c d e     Score     57
|   |   | | |     Nbrk       2
A B C D E F G     RMS     1.96
```

	A	B	C	D	E	F	G
a	**7**	5	9	2	1	0	0
b	2	9	**12**	9	7	2	0
c	1	2	2	10	**12**	8	2
d	0	1	1	2	2	**13**	7
e	0	0	0	0	1	2	**13**

```
a b - - c d e     Score     91
| |     | | |     Nbrk       1
A B C D E F G     RMS     0.65
```

	A	B	C	D	E	F	G
a	**19**	4	4	1	1	0	0
b	4	**16**	16	4	4	1	0
c	1	4	4	14	**18**	4	1
d	0	1	1	4	4	**19**	4
e	0	0	0	1	1	4	**19**

```
a b - - c d e     Score    100
| |     | | |     Nbrk       1
A B C D E F G     RMS     0.23
```

	A	B	C	D	E	F	G
a	**20**	4	3	1	1	0	0
b	4	**20**	12	4	4	1	0
c	1	4	4	11	**20**	4	1
d	0	1	1	4	4	**20**	4
e	0	0	0	1	1	4	**20**

Fig. 1. How pairwise structural alignment works. This schematic of our method of structural alignment is to be read from top to bottom. At the top are two highly simplified structures (ABCDEFG and abcde) in an arbitrary, initial orientation. An initial equivalence is chosen, based on matching the ends of the two structures. Using this equivalence, we can least-squares superimpose the two molecules (giving an RMS deviation in corresponding atoms of 1.96 Å, upper-middle). Then, based on relative positioning of the molecules determined from the fit, we calculate the distance, d_{ij}, between every atom i in the first structure and every atom j in the second structure. Each distance is transformed into a similarity value S_{ij} to form the similarity matrix shown at the upper-middle-right, ($S_{ij} = M/[1 + (d_{ij}/d_o)^2]$, where $M = 20$ and $d_o = 2.24$ Å). In the initial orientation atom "a" is close to atom "A" and even closer to atom "C," and this is reflected in the S_{ij} matrix values. Dynamic programming chooses the pairs indicated by the boldface S_{ij} entries. The score for this selection is the sum of the S_{ij} values of the selected pairs less the gap penalty for each chain break (nbrk). Using a default gap penalty of 10 ($M/2$), the score is $7 + 12 + 12 + 13 + 13 - 10 - 10$, for the S_{ij} matrix at the upper-middle-right. The pairs chosen by dynamic programming give a new set of equivalences shown in the lower-middle. These are used to do a second least-squares fit (giving an RMS of 0.65 Å). A new similarity matrix S_{ij} can now be calculated (shown at the lower-middle-right), and dynamic programming again used to find new equivalences. Finally, at the bottom we see that these equivalences give a perfect match, so a final cycle of dynamic programming does not change the alignment. The iteration has converged on an alignment.

rate on this by taking into account residue exposure and side-chain orientation, specifically by using beta carbons ("Cβ") or weighting according to the relative orientation of side-chain vectors ("Cα-Cβ"). A final elaboration allows the gap penalties to vary with position in the structure, so that it is more difficult to introduce breaks in helices and strands than in loops ("var. gap").

Using scop to assess our algorithm: A "meta-method"

Objective ways for assessing the quality and significance of our alignments are the key points that distinguishes what we do here from previous approaches towards automatic structural alignment. Our attention to validating our procedure against objective external standards is in a sense a "meta-method"—a method for evaluating a method.

To assess sensitivity, we checked our procedure against the entire scop database. This consists of hundreds of thousands of relationships between the ~8,000 protein domains of known structure. However, many of these relationships are trivial (e.g., same protein in different liganded states) or can readily be derived from sequence homology. The non-trivial relationships are evident only after clustering all the domains on the basis sequence. The current version of scop (1.32) contains 941 unique domains at a 40% identity cutoff (Brenner et al., 1995, 1997). Of the 441,330 possible pairs of these domains (940 × 939/2), 2,107 (~0.5%) are in the same scop superfamily and therefore have a similar 3D structure. These 2,107 pairs were what we tested our procedure against.

To check how accurate the alignments produced by our procedure were, we compared them against manual alignments published in the literature (making sure that these alignments were really done by hand and not a product of another computer algorithm). This was done in a most straightforward fashion, by optimally "aligning" the automatically generated alignments en bloc against the literature alignments and then counting the mismatches in the "core regions" (see Methods). This protocol is a much more objective test than simply inspecting the automatically produced alignments to see whether they "look" reasonable. In that situation it is possible to be either wittingly or unwittingly biased in favor of the program's alignments.

Overall "sensitivity" in finding the scop pairs

We ran our structural alignment program against all 2,107 of the scop pairs. Each comparison gave a value for the number of residues matched (N) and the RMS deviation in alpha-carbon positions after doing a least-squares fit with these N residues (the "RMS"). Our overall results are shown in Figure 2 through plotting RMS versus N for each scop pair. There is a fairly wide spread in the values for both RMS (2.66 ± 0.77 Å) and N (98 ± 57), but it is possible to approximately separate the successful matches (low RMS) from the unsuccessful matches (high RMS) by the demarcation line RMS = $4(N + 135)/225$. This sloping line indicates that a match with a higher RMS value can be more significant than one with a lower RMS if there are more residues in the first match, as is to be expected. Based on the demarcation line it is convenient to define a normalized RMS: RMS′ = 225 RMS/(N + 135). As shown in Figure 2b, plotting this quantity against N now gives a flat demarcation line of RMS′ = 4 Å (nearly the same as the distance between alpha carbons). Note that for an approximately average match of 90 residues, RMS′ is the same as RMS (and that both quantities agree to within 10% for N between 70 and 110 residues).

Figure 3 shows the distributions of RMS and RMS′ values. Both distributions have very similar means (2.66 and 2.68 Å) and standard deviations (0.77 and 0.87 Å). The normalized distribution has a sharp falloff for RMS′ greater than 4 Å, justifying this as a criterion for a significant match.

About 15% of the scop pairs (313 of 2,107) have some (marginal) sequence similarity (as indicated by a FASTA e-value less than 0.01, see legend to Fig. 2). All of these pairs are below the 4 Å RMS′ demarcation line, and collectively, they have a lower average RMS (1.9 Å) and a higher average N (129) than the other pairs,

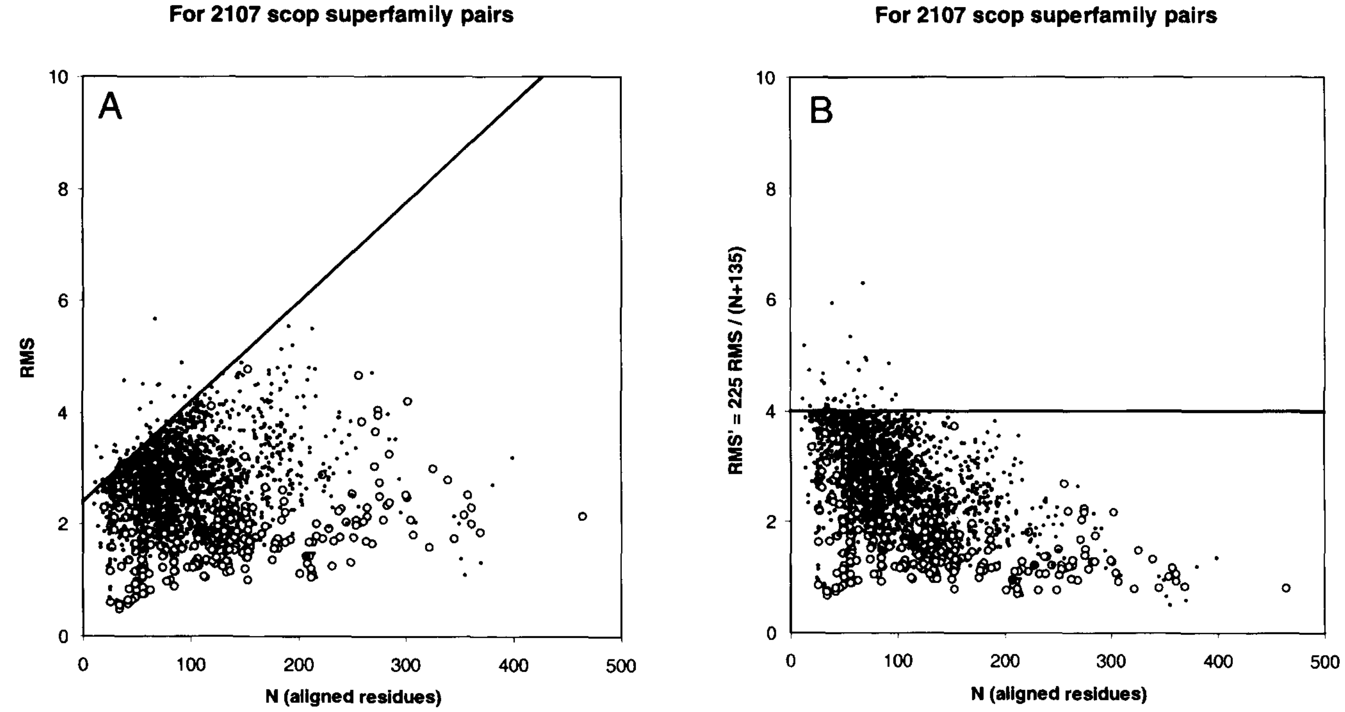


Fig. 2. Overall performance on the scop superfamily pairs. This figure shows the overall performance of our structural alignment algorithm on the 2,107 scop superfamily pairs. Part (**A**) shows a plot of RMS versus the number of residues matched N for each of the pairs. A demarcation line separating good matches from bad ones is drawn as RMS = $4(N + 135)/225$. Each pair that has some sequence similarity is indicated by an open circle. Clearly, these pairs tend to have somewhat closer structural matches. Sequence similarity was determined by doing an all-versus-all sequence comparison of the 941 scop domains using the FASTA program (with a k-tup value of 1) (Pearson & Lipman, 1988). An e-value for a pair less than 0.01 was taken to indicate significant sequence similarity with an expected false positive error rate of 1% (Brenner et al., 1995; Pearson, 1996). Note that none of the 941 domain structures in the 2,107 scop superfamily pairs has sequence identity greater than 40%, so the sequence similarity found by FASTA is, by definition, somewhat marginal. Part (**B**) is similar to part (**A**) but now a plot of the normalized RMS' vs. N is shown for the same pairs (RMS' = $225\text{RMS}/(N + 135)$). The demarcation line is now RMS' = 4 Å.

indicating that sequence similarity is related to structural similarity even this close to the "twilight zone."

Using a normalized RMS' threshold of 4 Å, we find that only 32 of the 2,107 pairs are outliers, less than 2%. These results were obtained using our "optimized" protocol that starts from a number of points and uses a variety of different parameter settings (see Methods). However, 1,762 of the pairs (~84%) could be found with just a single primary search method (Cβ). Of the remaining pairs, 313 (15%) could be found through application of multiple search strategies, leaving 32 pairs (1.5%) that we could not find at all.

Protein structure features that fooled the method, and why

We investigated in detail the 32 outliers that the program missed completely, trying to identify the types of protein structures that were fooling the program. (In this analysis we also looked at an additional 37 pairs where the match was slightly better than our 4 Å RMS' threshold but for which the number of aligned residues was less than 40% of the length of the smaller protein.) A number of these "bad pairs" represent unusual residue selections in the scop database; for instance, the scop pair with identifiers d1ggta1 and d1cdcb_ associates, respectively, a full immunoglobulin variable domain with a strangely shaped immunoglobulin fragment (see Methods for scop id syntax). Four of the seven worst failures involve the protein with scop identifier d1dhx__, which is an all-β animal virus-coat protein.

A number of the difficult to align pairs had circular permutations in their similar structural elements. For instance, the scop pair with identifiers d1scs__ and d1xnb__ consists of two proteins that share the same all-β, concanavalin-A fold but differ in connectivity, having a circular permutation of strands in the central sheet. As our alignment program was not designed to handle circular permutations, its difficulty with this pair is understandable. The complexities of handling topology changes in alignment has been discussed previously (Orengo et al., 1995).

Finally, we found a number of interesting cases where the structures were considered similar in scop because they shared a special structural feature rather than an overall similarity in shape. The scop pair shown in Figure 6 (d1dpga2 and d1gdlo2) represents such an instance. Both domains in the pair are considered to share the same superfamily fold, that of the C-terminal domain of glyceraldehyde-3-phosphate dehydrogenase. However, they only have a small amount of common structure, which is only remotely alignable—in particular, a four-stranded sheet with two helices packed on one face. Thus, in terms of the raw score used by the program (i.e., the average closeness of Cα atoms), the domains could not be matched well. However, they are grouped together in scop because both share a unique type of connectivity between the helices and strands, involving a rare type of loop "cross-over."

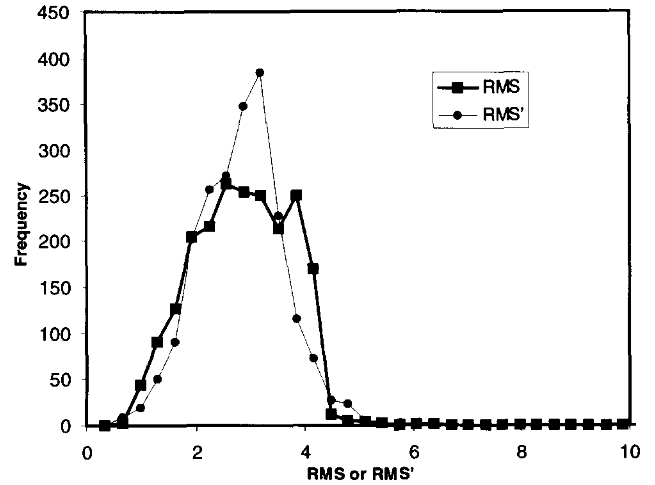


Fig. 3. Distribution of RMS values on the scop pairs. This figure shows the distribution of RMS and RMS′ values resulting from aligning each of the 2,107 scop superfamily pairs.

Moreover, this special cross-over occurs at the "heart" of these proteins, participating in the active site, and occurs in further proteins grouped in the same superfamily (e.g., d1dih_2).

Detailed accuracy of the alignments

For the remaining scop pairs that could be aligned with acceptable RMS values, we tried to assess the quality of their alignments in detail. To this end, we compared a set of nine automatically generated multiple alignments, based on portions of the scop superfamilies (involving 40 structures in total), with corresponding manual alignments culled from the literature. Our overall results, in terms of mismatches for each set, are shown in Table 1. Our selection of test cases represents a wide variety of protein structures: all-α (globins), all-β (immunoglobulins, plastocyanin-azurin), α/β (dihydrofolate reductase family), structures with large conformational changes in addition to evolutionary changes (adenylate kinases), and structures with large inserts (Gal6-papain). As was the case with the sensitivity analysis, we found overall that the basic method, minimizing Cα distance, works. However, it has some trouble with beta-sheet proteins.

Our results are shown in greater detail in Figure 7, which shows the automatically generated alignments of three well-known protein families: the all-alpha globins, the all-beta immunoglobulins, and the alpha/beta dihydrofolate reductase family. Mismatches in the core regions are indicated. The globins and the dihydrofolate reductases are "easy" to align (Fig. 4). The basic procedure (Cα), as well as any of the variants, was able to generate the correct alignment.

The immunoglobulins are more problematical, especially with regard to aligning the constant and variable domains. As shown in Figures 5 and 7, all the variants of our algorithm will generate an alignment with an acceptable RMS, but the alignments differ in detail. In fact, the alignment that minimizes alpha-carbon distance looks deceptively correct and has the best overall RMS. However, it is clearly wrong, as it misaligns the conserved disulfide. A variant of our procedure that takes into account side-chain orientation and also uses variable gap penalties is necessary to get the alignment right. Aligning immunoglobulin constant and variable domains has proven difficult with other structural alignment methods (Taylor & Orengo, 1989). The difficulties we found for the immunoglobulins and other all-β proteins suggest that, in general, it is easier to misalign strands than helices unless one takes into account side-chain orientation.

Figure 8 shows how our simple approach toward multiple alignment, align to the "median structure" in the cluster, performs. Clearly, as one moves away from the median structure the alignment degrades. Nevertheless, the overall mismatch error rate is very low using this approach (Table 1), indicating it is probably sufficient for the superfamily size currently in scop. [However, in the future this could change (see Methods).]

Table 1. *Comparison of automatically generated multiple alignments vs. manual "gold standards"*[a]

Protein family	Num. struct.	Num. comp.	Mismatches	Scop s. fam.	Comment on structures	Method
1 Plastocyanin/azurin	2	118	2	2.05.1	All-β	Cα
2 Immunoglobulin VL-Fc (V-set + C1-set)	2	72	6	2.01.1	All-β	Cα-C β + var. gap
3 Cysteine proteinases (Gal6-Papain)	2	214	2	4.03.1	α + β with large insertions	Cα
4 C-type lectins	2	212	0	4.77.1	α + β (mostly β)	Cα
5 P-loop containing NTP hydrolases (ADK)	3	534	0	3.21.1	α/β with large conf. change	Cα
6 Immunoglobulin V-frame (V-set + I-set)	4	184	4	2.01.1	All-β (includes telokin)	Cβ + var. gap
7 Dihydrofolate reductases	4	436	1	3.46.1	α/β	Cα
8 Globins	8	805	18	1.01.1	All-α	Cα + var. gap
9 Immunoglobulin V-set (just VL domains)	13	1,183	11	2.01.1	All-β	Cβ

[a] The table shows summary statistics derived from comparing nine automatically generated alignments to manual, "gold-standard" alignments culled from the literature. These alignments are meant to correspond to as varied a selection of scop superfamilies as possible, given the limitations of the data in the literature. A detailed explanation of the statistics follows: Column "num. struct." gives the number of structures involved in the alignment. Column "num. comp." gives the number of comparisons done in comparing to the manual alignment. This is just the number of core positions times the number of structures. Column "mismatches" gives the number of mismatches compared to the hand alignment (which should be considered relative to the total number of comparisons). Column "scop s.fam." gives the scop superfamily that the alignment was generated from. Column "method" tells whether the basic method (Cα) or a variant was used in generating the alignment. Alignment 1 is from Chothia and Lesk (1982); 2, Lesk and Chothia (1982); 3, Joshua-Tor et al. (1995); 4, Graves et al. (1994); 5, Gerstein et al. (1993); 6, Harpaz and Chothia (1994) and Leahy et al. (1992); 7, Gerstein et al. (1994); 8, Lesk and Chothia (1980); 9, Chothia and Lesk (1987). All of the "gold-standard" alignments were done truly manually (i.e., not by using a different computer algorithm).

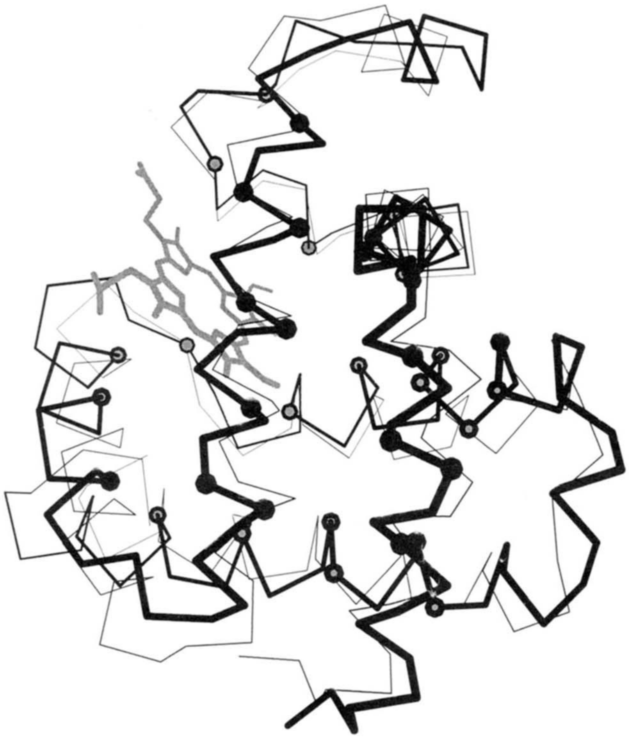


Fig. 4. An easy to align pair (globins). This figure shows a sample structural alignment of a pair of globins (d1mbd__ and d1ecd__; see Methods for a discussion of the scop identifier conventions). The aligned positions are indicated by small, gray CPK spheres. This alignment is "easy" in the sense that it is obtainable from either the basic algorithm ($C\alpha$) or any variant (e.g., $C\beta$), and that there are very few mismatches compared to the hand alignment taken from the literature. See Figure 7b for another view of this alignment.

Discussion and conclusion

Summary

We have described how we applied an automatic structural alignment method, on a large scale, to the proteins in the manually constructed scop classification of fold families. Comparing our automatic alignments against manual standards has allowed us to get a relatively unbiased assessment of how a uniformly applied computer procedure compares with human experts, both in terms of overall sensitivity and detailed accuracy. In a sense, our program has acted as a foil to expose the subtleties of protein structural similarity.

Testing the program against objective standards has allowed us to refine it (taking into account such things as side-chain orientation and exposure, variable gap penalties, and a more comprehensive search), measurably increasing our overall sensitivity (so that we could eventually find 99% of the scop pairs). We have also demonstrated a simple yet effective scheme for generating multiple structural alignments based on our pairwise alignments.

Easy, hard, and impossible pairs

Proteins in the same scop superfamily are considered to have evolved from a common ancestor (Murzin et al., 1995). Such an evolutionary relationship does not necessarily imply conservation of sequence or structure. All that is needed is that the proteins are considered to have more in common (in terms of sequence, structure and/or function) than would be expected to arise independently or by convergent evolution. Thus, although some of the 2,107 superfamily pairs have significant sequence similarity (~16%; Brenner et al., 1997), others do not.

We find that our method partitions the evolutionary relationships in scop roughly into three categories based on alignability: easy to align, hard to align, and impossible to align. In the first category are proteins, such as the globins (Fig. 4), which can be aligned correctly by our basic method ($C\alpha$) or any of the variants. In the next category are proteins, such as the immunoglobulins (Fig. 5), that need a modified method (e.g., $C\beta$) for successful alignment. This is necessary either because the basic method cannot find an alignment with an acceptable RMS or because, even though it finds an alignment with a good RMS, it does not get this alignment completely right. As a rough rule, proteins in this second category tend to have more sheet structure than helical structure, probably because of the greater structural variability allowed in strands than helices and also because (without considering side-chain orientation) it is easier to misalign a strand by one residue.

Finally, in the last category are the ~1.5% of the scop pairs that we could not align at all by the basic methods or any variants (Fig. 6). Our difficulty with a number of these pairs can be understood because a specific protein–structure "feature," such as crossed loops, is used as the basis for a resemblance, rather than simply the similarity in backbone structure.

Directions: Statistical significance and sequence applications

The distinction between easy, hard, and impossible to align pairs is obviously related to the statistical significance of a given structural similarity, i.e., how good the match is compared to random expectation (the P-value). Statistical significance for structural alignment can be evaluated in a similar fashion as for normal sequence alignment (Altschul et al., 1994; Pearson, 1996), by deriving statistical models from the results of all possible pairwise comparisons. Work in this direction is on-going, and we have recently derived a formula for the significance of a structural alignment (Levitt & Gerstein, 1998). The threshold of $RMS' = 4$ Å used here corresponds approximately to significance level (or P-value) of 0.01.

In any case, as we have been able to find a reasonable match for almost all the scop superfamily pairs (98.5%), our alignments are expected to be useful in many applications, ranging from testing sequence alignment and fold recognition algorithms to the defining appropriate structural "modules" for searching the genome (Brenner et al., 1997; Gerstein, 1997; Gerstein & Levitt, 1997; Sonnhammer et al., 1997).

Methods

Data

Structures were taken from the Protein Data Bank (PDB; Bernstein et al., 1977). Version 1.32 of the scop fold classification was used (May 96) (Murzin et al., 1995; Brenner et al., 1996; Hubbard et al., 1997). This includes a number of structural similarities that were not in the PDB (i.e., they were taken from the literature). It also has some proteins that have multi-chain "domains." These and other special cases were removed from the database. Each of the do-

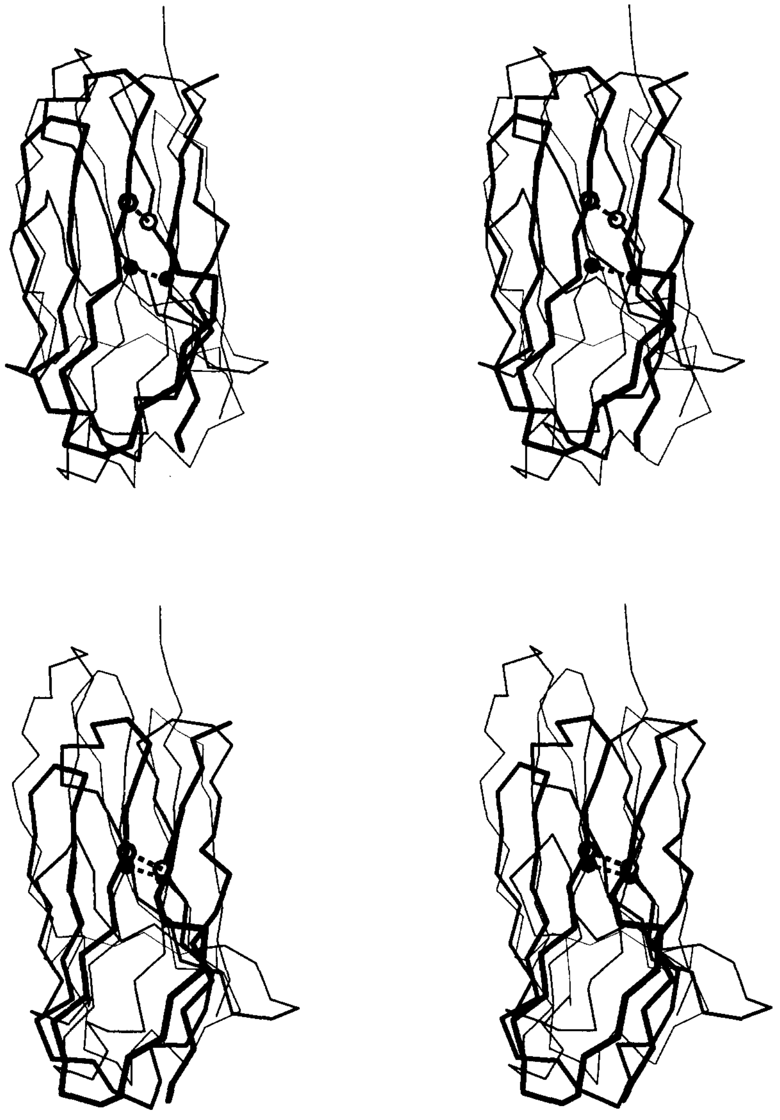


Fig. 5. A harder to align pair (immunoglobulins). This figure shows an alignment of immunoglobulin light-chain variable domain (d7fabl2) with an immunoglobulin constant domain (d1reia1). One can readily "match" this pair with the basic method (Cα) or any of the variants (in the sense that one can get a good RMS′ value). However, it is deceptively difficult to get the correct alignment in detail. The alignment from the basic method, just matching Cα atoms, is shown on the right. It gets a reasonable RMS from matching all the atoms and after elimination (see table below). However, it is clearly wrong because it misaligns the conserved disulfide (shown by the CPK spheres in the figure). In fact, comparison with the hand alignment shown in Figure 7c indicates that every strand is slightly misaligned, giving 28 mismatches in total. It is necessary to use a variant method, which takes into account sidechain orientation and variable gap penalties, to get an alignment that gets the disulfides right. This alignment is shown at the left.

Method	Variant (Cα-Cβ + var. gap)	Basic (Cα atoms)
Mismatches vs. hand (36 aligned so/72)	6	28
RMS from all equiv. Cαs (84)	4.0 Å	3.1 Å
RMS after elimination (best 36)	1.7 Å	2.0 Å

mains classified by scop is associated with a unique identifier, and these are used throughout this text. They have the following syntax: d1pdbcN, where "1pdb" is a PDB id, "c" is a chain identifier, and "*N*" describes if this is the first, second, or only domain in the chain. Thus, d1ggta1 is the first domain in the A chain of 1GGT.

The creators of scop have clustered the domains in the PDB on the basis of sequence identity (Brenner et al., 1995, 1997), using a procedure similar to that of Hobohm et al. (1992). At a sequence identity level of 40%, this procedure results in 941 sequences corresponding to the scop domains. These sequences contain 176 different superfamilies, which involve 2107 nontrivial pair relationships between the domains. (Only superfamily pairs were used here as they have a considerably closer and more certain relationship than fold pairs.)

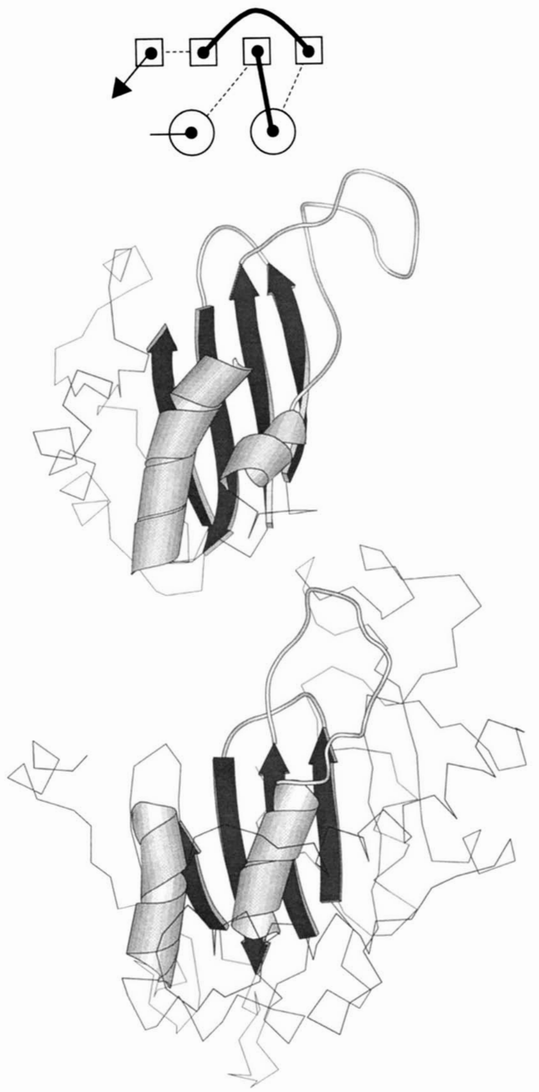


Fig. 6. A very hard to align pair (G3P dehydrogenase C-term. domain). This figure shows a scop pair that our program was not able to align at all. These structures (d1gd1o2 in the middle and d1dpga2 at the bottom) are considered to share the fold of the C-terminal domain glyceraldehyde-3-phosphate dehydrogenase. However, they have in common only a small core region of similar topology, consisting of a four-stranded sheet with two helices packed on a face. This is highlighted in the structures and indicated in the topology diagram at TOP. The structures are grouped together in scop principally because they share an unusual type of cross-over connection, joining the strands in the sheet. This connection is highlighted by a bold line in the topology diagram and a thick ribbon in the middle and bottom sub-figures. In both structures the crossed loops are inserted into the Rossmann-fold NAD(P)-binding domain in the same place, so they form an equivalent part of the active site. Furthermore, there is a third member of this scop superfamily (d1dih_2) that has a pair of cross-loops equivalently inserted into a Rossmann-fold–like domain.

The basic procedure, minimize Cα RMS

The basic procedure we use for pairwise structural alignment is based on iterative application of dynamic programming. As such, it is a simple generalization of Needleman-Wunsch sequence alignment (Needleman & Wunsch, 1971). The basic method is originally derived from the ALIGN program of G Cohen (Satow et al., 1987; Cohen, 1997) and has been applied to specific cases previously (Gerstein & Levitt, 1996). As shown in Figure 1, one starts with two structures in an arbitrary orientation. Then one computes all pairwise distances between each atom in the first structure and every atom in the second structure. This results in an *inter*-protein distance matrix where each entry d_{ij} corresponds to the distance between atom i in the first structure and atom j in the second one. This distance matrix can be converted into a similarity matrix S_{ij}, similar to the one used in sequence alignment, by application of the following formula:

$$S_{ij} = \frac{M}{1 + \left(\frac{d_{ij}}{d_o}\right)^2}.$$

Here, M is the maximum score of a match, which is arbitrarily chosen to be 20. d_o is the distance at which the similarity falls to half its maximum value (i.e., $d_{ij} = d_o \rightarrow S_{ij} = M/2$). It is taken here to be 2.24 Å—reflecting the intrinsic length scale of protein structural similarity. This is about midway between the length of a C–C bond (1.54 Å) and the usual distance between Cα atoms (3.8 Å).

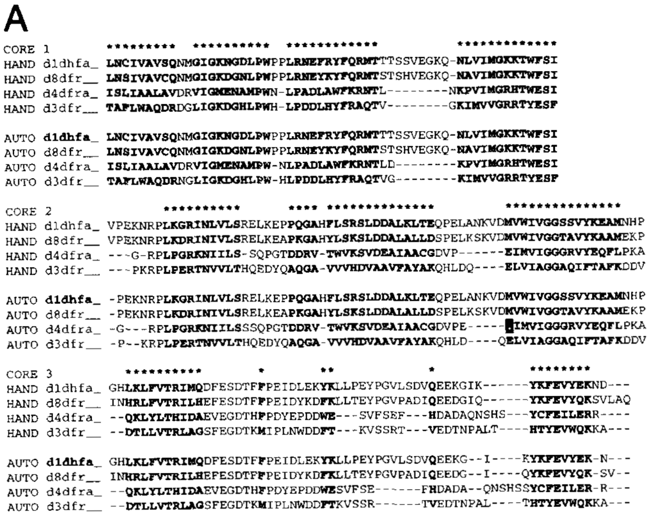
One applies dynamic programming to the similarity matrix to get equivalences. If this were normal sequence alignment, one would be finished at this point since dynamic programming gives the optimal equivalences. However, this is not the case for structural alignment. So one takes these equivalences and uses them to least-squares fit the first structure onto the second one (Kabsch, 1976). Then one repeats the procedure over and over, finding all pairwise distances and doing dynamic programming to get new equivalences, until it converges on the same set of equivalences.

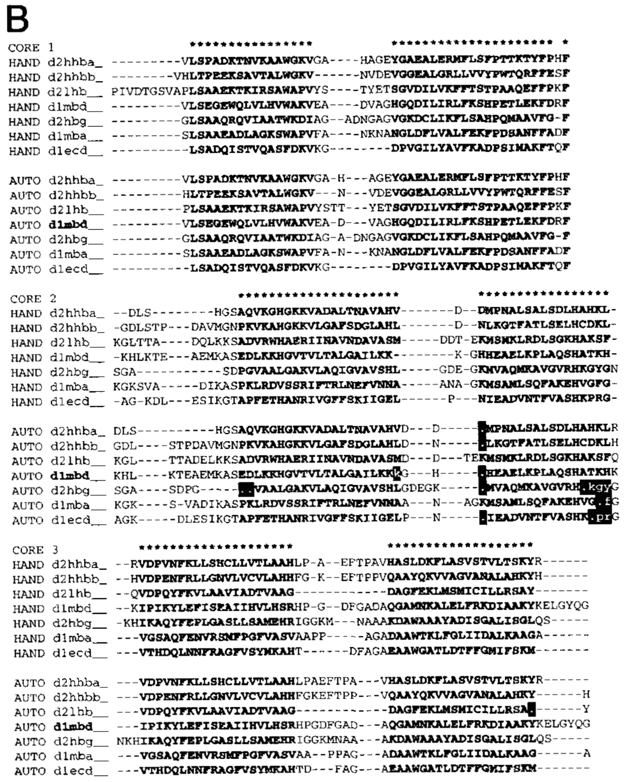
Basic search

In practice, the iteration is tried from a number of different starting points, and the one that gives the best score is taken. This score is calculated as the sum of the S_{ij} values of the selected equivalent pairs (i, j) from the dynamic programming less the penalty for each of the chain breaks or gaps. We use six starting alignments, giving different sets of initial equivalences: (1) align the beginnings of the two sequences, (2) align the midpoints, (3) align the ends, (4) align at a random point, (5) align using sequence identity, and (6) align using alpha angles. Most of these starting points were used previously (Subbiah et al., 1993; Laurents et al., 1994; Gerstein & Levitt, 1996). However, to correctly match all the scop pairs, we needed to modify our procedure as discussed below.

Side-chain orientation

An important improvement was taking into account sidechain orientation. This could simply be done by using Cβ rather than Cα atoms for the computation of distances d_{ij}. However, we sometimes used a more elaborate procedure (method CαCβ) where we multiplied each entry in the similarity matrix S_{ij} by a factor representing the relative orientation of the Cα-Cβ (or C = O) bonds [specifically exp(cos A), where A is the angle between the corresponding bond in each structure]. Taking into account side-chain orientation makes misalignments by one residue in helices and, especially, in strands more difficult. Misalignments by a single residue are not serious in terms of matching the overall fold but give nonsensical alignments in detail. For instance, in the case of strands they often lead to mismatching of hydrophobic and hydrophilic residues.





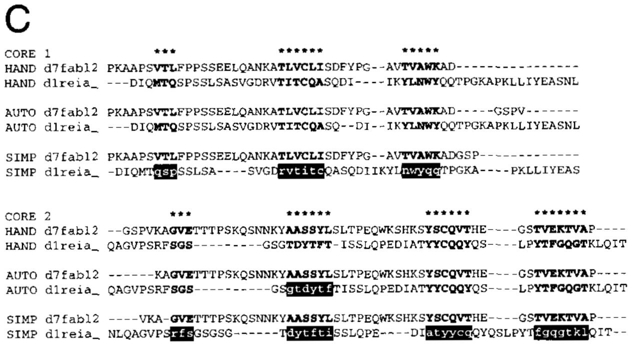


Fig. 7. Sample multiple alignments. This figure shows sample multiple alignments for three protein families. Part (a) shows one for the dihydrofolate reductase (DHFR) family; part (b), for the globin family; and part (c), for two immunoglobulins. For each family, in turn, two separate multiple alignments are shown: the one marked "hand" is a manually constructed "gold standard" taken directly from the literature and the one marked "auto" is automatically generated by our program. The hand alignments were taken from Lesk and Chothia (1982) for the immunoglobulins, Gerstein et al. (1994) for the dihydrofolate reductases, and Lesk and Chothia (1980) for the globins. The hand and auto alignments were aligned as blocks so that there are the fewest possible mismatches between them. Mismatches are scored only in the core alignable regions, marked by a "*" character in the "core" row. They are highlighted in the automatically generated alignment (by inverted text, changing case, and substituting "-" for "."). The DHFR alignment has one mismatch in total with d1dhfa_ as the central structure to which everything is aligned. The globin alignment has 18 mismatches with d1mbd__ as the central structure. For the immunoglobulins, a third alignment, beyond the hand and auto ones, marked "simp" is also shown. This is a result of using the basic method (Cα). It clearly gets the alignment wrong and a more complex method is necessary to get the correct alignment (Cα-Cβ + var. gap). See Figure 5 and the text for more details.

Exposure weighting

Another useful modification was to increase the weight of the aligned residues buried inside the protein relative to those on the surface. This was achieved through the following procedure: the accessible surface area (Lee & Richards, 1971) of each residue was determined (considering an all atom model). These areas (in square Angstroms) were used to assign weights $W(i)$ to each residue i according to the following scheme: 0.5 for the exposed residues (exposed area greater than 100 Å^2), 2.0 for the buried residues (exposed area less than 50 Å^2), and of 1.0 for the remaining residues. These weights were then used to modify the entries of the similarity matrix (S_{ij}) as follows: $S'_{ij} = W(i)W(j)S_{ij}$.

Secondary structure-dependent gap penalties

In the basic version of the method, the gap penalty is independent of gap size and normally taken to be half the score contribution of a perfectly matched pair (i.e., $M/2 = 10$). Because of the similarity between our structural alignment procedure and normal sequence alignment, it is possible to incorporate more complicated variable, position-dependent gap penalties into the alignment in a very straightforward fashion. Because we know the secondary structure of the two proteins we are aligning (e.g., from DSSP, Kabsch & Sander, 1983; or stride, Frishman & Argos, 1995) we can make it more difficult to introduce a gap at a position in a secondary structure (i.e., strand or helix). This is similar to *sequence* alignment methods that make the penalty for opening a gap depend on where it starts (Lesk et al., 1986; Smith & Smith, 1992; Vingron & Waterman, 1994). Other methods for structural alignment have also employed this approach (Zhu et al., 1992).

We derived specific values for the gap penalties by empirically testing them on a number of protein families. We found that as the gap opening penalty is decreased in secondary structure relative to that in loops and coils, one obviously increases the number of

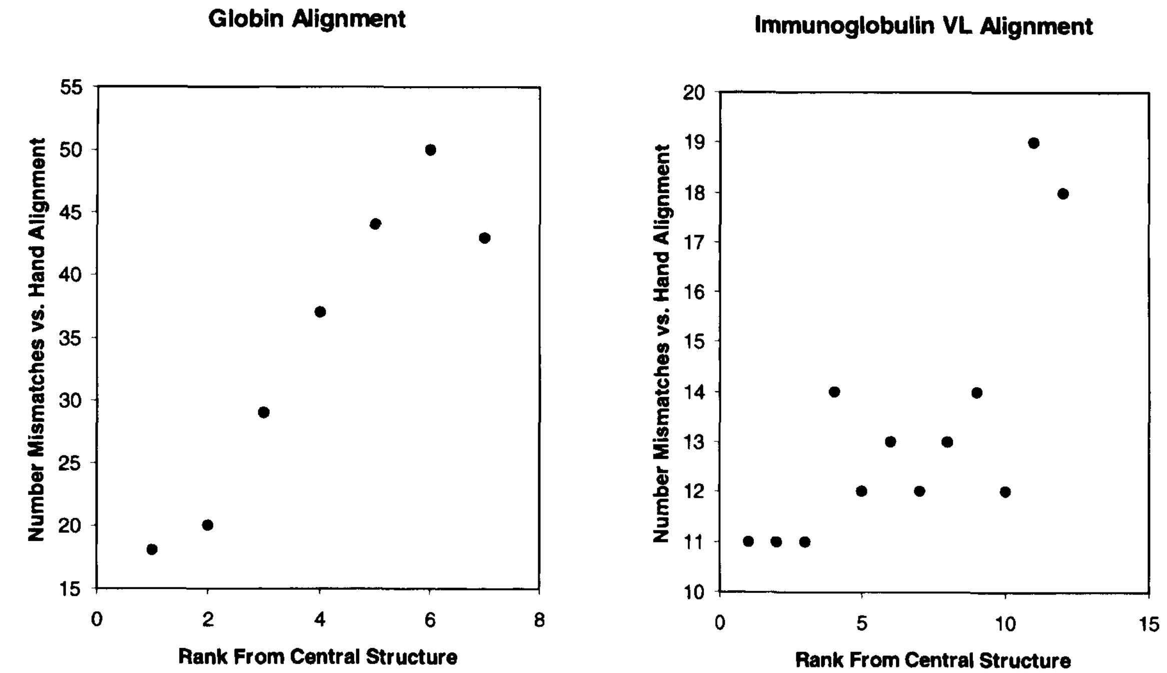


Fig. 8. Median structure and multiple-alignment quality. This table shows the how the quality of a multiple structural alignment decreases as one moves away from using the median structure as a basis for the alignment. Two families of structures are shown: immunoglobulin VL domains (all-β) and globins (all-α). For each family all possible pairwise alignments were done and then used to calculate the average distance (i.e., average RMS) between each structure and all the other structures. Because this distance will be smallest for structures near the cluster center, it can be used to rank each structure in terms of its proximity to the cluster center. Next, a multiple alignment was automatically generated based on a aligning all the structures in the family to a particular target structure. Every structure, in turn, was considered as the target. As described in the text, our automatically generated alignments were compared with manually generated "gold-standard" alignments, and the total number of comparisons and mismatches at core positions were tabulated. As we consider target structures farther away from the "center of the structure cluster" (in the RMS sense discussed above) the number of mismatches increases. This is true for both the highly diverged globin alignment and the less-diverged immunoglobulin alignment.

spurious gaps in strands and helices. This suggests that very high gap penalties in strands and helices might work well. However, we also found that such high gap penalties make it more difficult to align secondary structural elements (which often vary slightly in size); in fact, a penalty that is too high leads to completely mismatching secondary structures. (For instance, instead of aligning two helices of slightly different size through introducing a gap into the longer helix, the program might introduce many gaps into a loop preceding one helix and align this helix against a loop and the second against the introduced gaps.) The specific values we chose are a compromise between these two competing effects. We always set the gap extension penalty to be a small constant value (0.025 M). We arranged the gap opening penalties for each structure into a vector $\alpha(k)$, indexed by the sequence position i or j. Initially, the $\alpha(k)$ values were set to 2 in sheets and helices, and 1, otherwise. $\alpha(k)$ is then smoothed (by convolution with a Gaussian having weights 1,3,8,3,1) and re-scaled so that the overall average gap penalty $\langle\alpha(k)\rangle$ is half the maximum match score M.

Refinements to the search

When comparing structures of different size it was sometimes advantageous to split the larger structure into pieces. Here we used three pieces: the first half, the middle half, and the second half. Because each of the structures in the set of 941 scop domains was only a single domain, this trimming was only used for 82 pairs out of the total of 2,107 (3.9%). Of these 82 comparisons, 50 lead to a successful match. For protein structures that have not been separated into domains, this splitting is most useful for structures with internal symmetry and duplication, such as calmodulin, and for structures that had a small strong similarity in the midst of larger overall similarity that was not as well defined.

The best search strategy consisted of the following five steps: (1) use Cβ atoms; (2) use Cα atoms; (3) use Cβ atoms with exposure weights; (4) use Cα atoms with exposure weights; and (5) try three-way splitting of the longer chain with Cβ atoms and exposure weights. The search is stopped after any step that does not fail, where failure is defined as not being able to get an RMS′ score (defined in the results) less than 4 Å. This strategy is somewhat arbitrary, and other protocols give similar results (e.g., Cα atoms followed by Cα atoms with variable gap penalties, followed by Cβ atoms). The important point is that starting from multiple starting points and using a variety of different definitions for the similarity matrix helped. Furthermore, although it is true that the search strategy can be adjusted, once the parameters are chosen, the alignments are all generated automatically, and the results reported here are of a completely automated run.

Elimination to determine a core structure

After determining an alignment, we "refined" the RMS by eliminating the worst fitting pairs of equivalenced atoms and then refitting to get a new RMS, in a similar fashion to the core-finding procedure in Gerstein and Altman (1995a, 1995b). This refinement is necessary as the dynamic programming tries to match as many residues as possible (i.e., it is a global as opposed to a local method). In doing the elimination, we do not change the equivalences but merely eliminate those pairs with the worst individual deviation in atom position.

The threshold for stopping the elimination is somewhat arbitrary. We tried a variety of approaches (absolute threshold, "throwing-out" a given fraction of the atoms, etc.). The scheme we settled on involves eliminating the pair of equivalenced atoms with the largest interatomic distances so long as the following criteria are satisfied: (i) The chosen pair must be adjacent to a chain break (or chain ends), which ensures that the elimination procedure does not increase the number of gaps. (ii) The pair to be eliminated has to have a separation d_{ij} greater than 3.8 Å, which is the distance between adjacent Cα atoms along the polypeptide chain. (iii) Fewer than half the initial pairs have been eliminated. (iv) There are more than 20 remaining pairs. (v) If there are fewer than 50 matches, the RMS′ must exceed 4 Å, which prevents the elimination procedure from generating very short segments that match well.

This elimination scheme performed well in that the lengths of matched regions (N) were not excessively shortened, while at the same time the RMS deviations were reduced considerably. That is, the average RMS drops from 4.64 Å to 2.66 Å, while the average N drops only from 123 to 98. It is also interesting to note that for the 2,107 scop pairs, elimination was stopped 82% of the time for criteria (ii) (see above), 5% of the time for criteria (iii), 1% for (iv), and 12% for (v).

Multiple structural alignment

We formed multiple structural alignments by combining all possible pairwise alignments for a given collection of structures. From all the pairwise alignments, we picked the structure that is on average closest to all other structures. This is in a sense the "median" structure within the "cluster" of all the structures. We then form a multiple alignment by aligning all the other structures to this median structure and consistently combining their alignments. Tests given show that aligning all the structures to non-median structures does not work as well (Fig. 8).

This procedure is somewhat simpler than the usual approach to multiple alignment, for both sequences and structures (Thompson et al., 1994; Taylor et al., 1994; Gotoh, 1996; Gusfield, 1997), which proceeds by agglomerative clustering. Often this involves forming a consensus between the closest pairs and then using this in subsequent steps. We felt our simple approach was adequate for the task at hand, as none of our multiple alignments involved a large number of structures (i.e., not more than 15 and usually only around 4). However, it would probably not give optimum results on a much larger number of objects (>100 as is often common in multiple *sequence* alignment).

Comparison of multiple alignments (manual versus automatic)

We compared our automatically generated multiple alignments against manual ones by "aligning" them en bloc by dynamic programming so as to minimize the total mismatches. We only count mismatches in structurally conserved core regions, as other regions of the protein structure, particularly some surface loops, are impossible to align correctly. Core regions are often explicitly indicated in the literature alignment. When this is the case, we used the literature definition. Otherwise, we used the elimination procedure described above and the somewhat more general strategies in Gerstein and Altman (1995a) to automatically determine a core.

Availability of results over the Internet

We will make available over the Internet at the following URL alignments of the scop superfamilies plus a table of scores (e.g., RMS and N) for each alignment: http://bioinfo.mbb.yale.edu/Align. There are also plans to provide an alignment server to align two arbitrary input structures.

Acknowledgments

We thank A. Murzin, S. Brenner, C. Chothia, T. Johnson, and W. Krebs for helpful conversations and/or reading the manuscript. MG acknowledges the NSF for support (Grant DBI-9723182) and ML, the DOE (Grant DE-FG03-95ER62135).

References

Altschul SF, Boguski MS, Gish W, Wootton JC. 1994. Issues in searching molecular sequence databases [Review]. *Nat Genet 6*:119–129.

Argos P, Rossmann MG. 1979. Structural comparisons of heme binding proteins. *Biochemistry 18*:4951–4960.

Artymiuk PJ, Mitchell EM, Rice DW, Willett P. 1989. Searching techniques for databases of protein structures. *J Inform Sci 15*:287–298.

Artymiuk PJ, Poirrette AR, Rice DW, Willett P. 1997. A polymerase I palm in adenylyl cyclase? [letter] [In Process Citation]. *Nature 388*:33–34.

Bernstein FC, Koetzle TF, Williams GJB, Meyer EF Jr, Brice MD, Rodgers JR, Kennard O, Shimanouchi T, Tasumi M. 1977. The protein data bank: A computer-based archival file for macromolecular structures. *J Mol Biol 112*:535–542.

Brenner S, Chothia C, Hubbard T. 1997. Assessing sequence comparison methods. *Proc Natl Acad Sci USA*. Forthcoming.

Brenner S, Chothia C, Hubbard TJP, Murzin AG. 1996. Understanding protein structure: Using scop for fold interpretation. *Methods Enzymol 266*:635–642.

Brenner S, Hubbard T, Murzin A, Chothia C. 1995. Gene duplication in *H. Influenzae*. *Nature 378*:140.

Bryant SH, Madej T, Janin J, Liu Y, Ruoho A, Zhang G, Hurley JH. 1997. A polymerase I palm in adenylyl cyclase? A Reply. *Nature 388*:34.

Chothia C, Gerstein M. 1997. Protein evolution. How far can sequences diverge? *Nature 385*:579–581.

Chothia C, Lesk AM. 1982. The evolution of proteins formed by beta sheets I. Plastocyanin and azurin. *J Mol Biol 160*:309–323.

Chothia C, Lesk AM. 1987. Canonical structures for the hypervariable regions of the immunoglobulins. *J Mol Biol 196*:901–917.

Cohen GH. 1997. ALIGN: A program to superimpose protein coordinates, accounting for insertions and deletions. *J Appl Crystallogr*. Forthcoming.

Doolittle RF. 1987. *Of Urfs and Orfs*. Mill Valley, CA: University Science Books.

Falicov A, Cohen FE. 1996. A surface of minimum area metric for the structural comparison of proteins. *J Mol Biol 258*:871–892.

Feng ZK, Sippl MJ. 1996. Optimum superimposition of protein structures: Ambiguities and implications. *Fold Des 1*:123–32.

Frishman D, Argos P. 1995. Knowledge-based secondary structure assignment. *Proteins 23*:566–579.

Gerstein M. 1997. A structural census of genomes: Comparing bacterial, eukaryotic and archaeal genomes in terms of protein structure. *J Mol Biol 274*:562–576.

Gerstein M, Altman R. 1995a. Average core structures and variability measures for protein families: Application to the immunoglobulins. *J Mol Biol 251*:161–175.

Gerstein M, Altman R. 1995b. A structurally invariant core for the globins. *CABIOS 11*:633–644.

Gerstein M, Levitt M. 1996. Using terative dynamic programming to obtain accurate pairwise and multiple alignments of protein structures. In: *Proc Fourth Int Conf on Intell Sys Mol Biol*. Menlo Park, CA: AAAI Press. pp 59–67.

Gerstein M, Levitt M. 1997. A structural census of the current population of protein sequences. *Proc Natl Acad Sci USA*. Forthcoming.

Gerstein M, Schulz G, Chothia C. 1993. Domain closure in adenylate kinase: Joints on either side of two helices close like neighboring fingers. *J Mol Biol 229*:494–501.

Gerstein M, Sonnhammer E, Chothia C. 1994. Volume changes on protein evolution. *J Mol Biol 236*:1067–1078.

Gibrat JF, Madej T, Bryant SH. 1996. Surprising similarities in structure comparison. *Curr Opin Struct Biol 6*:377–385.

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

Godzik A, Skolnick J. 1994. Flexible algorithm for direct multiple alignment of protein structures and sequences. *CABIOS 10*:587–596.

Gotoh O. 1996. Significant improvement in accuracy of multiple protein sequence alignments by iterative refinement as assessed by reference to structural alignments. *J Mol Biol 264*:823–838.

Graves BJ, Crowther RL, Chandran C, Rumberger JM, Li S, Huang KS, Presky DH, Familletti PC, Wolitzky BA, Burns DK. 1994. Insight into E-selectin/ligand interaction from the crystal structure and mutagenesis of the lec/EGF domains. *Nature 367*:532–538.

Gribskov M, Devereux J. 1992. *Sequence analysis primer*. New York: Oxford University Press.

Grindley HM, Artymiuk PJ, Rice DW, Willett P. 1993. Identification of tertiary structure resemblance in proteins using a maximal common subgraph isomorphism algorithm. *J Mol Biol 229*:707–721.

Gusfield D. 1997. *Algorithms on strings, trees and sequences*. New York: Cambridge University Press.

Harpaz Y, Chothia C. 1994. Many of the immunoglobulin superfamily domains in cell adhesion molecules and surface receptors belong to a new structural set which is close to that containing variable domains. *J Mol Biol 238*:528–539.

Hobohm W, Scharf M, Schneider R, Sander C. 1992. Selection of representative protein data sets. *Protein Sci 1*:409–417.

Hogue CW, Ohkawa H, Bryant SH. 1996. A dynamic look at structures: WWW-entrez and the molecular modeling database. *Trends Biochem Sci 21*:226–229.

Holm L, Sander C. 1993a. Protein structure comparison by alignment of distance matrices. *J Mol Biol 233*:123–128.

Holm L, Sander C. 1993b. Structural alignment of globins, phycocyanins and colicin A. *FEBS Lett 315*:301–306.

Holm L, Sander C. 1994. The FSSP database of structurally aligned protein fold families. *Nucleic Acid Res 22*:3600–3609.

Holm L, Sander C. 1996. Mapping the protein universe. *Science 273*:595–602.

Holm L, Sander C. 1997. New structure—Novel fold? *Structure 5*:165–171.

Hubbard TJP, Murzin AG, Brenner SE, Chothia C. 1997. SCOP: A structural classification of proteins database. *Nucleic Acids Res 25*:236–239.

Joshua-Tor L, Xu HE, Johnston SA, Rees DC. 1995. Crystal structure of a conserved protease that binds DNA: The bleomycin hydrolase, Gal6. *Science 269*:945–950.

Kabsch W. 1976. A solution for the best rotation to relate two sets of vectors. *Acta Crystalogr A32*:922–923.

Kabsch W, Sander C. 1983. Dictionary of protein secondary structure: Pattern recognition of hydrogen-bonded and geometrical features. *Biopolymers 22*:2577–2637.

Laurents DV, Subbiah S, Levitt M. 1994. Different protein sequences can give rise to highly similar folds through different stabilizing interactions. *Protein Sci 3*:1938–1944.

Leahy DJ, Axel R, Hendrickson WA. 1992. Crystal structure of a soluble form of the human T cell coreceptor CD8 at 2.6 Å resolution. *Cell 68*:1145–1162.

Lee B, Richards FM. 1971. The interpretation of protein structures: Estimation of static accessibility. *J Mol Biol 55*:379–400.

Lesk AM, Chothia C. 1982. Evolution of proteins formed by β-Sheets II. The core of the immunoglobulin domains. *J Mol Biol 160*:325–342.

Lesk AM, Chothia CH. 1980. How different amino acid sequences determine similar protein structures: The structure and evolutionary dynamics of the globins. *J Mol Biol 136*:225–270.

Lesk AM, Levitt M, Chothia C. 1986. Alignment of amino acid sequences of distantly related proteins using variable gap penalties. *Protein Eng 1*:77–78.

Levitt M, Gerstein M. 1998. A unified statistical framework for sequence comparison and structure comparison. *Proc Natl Acad Sci USA*. In press.

Murzin A, Brenner SE, Hubbard T, Chothia C. 1995. SCOP: A structural classification of proteins for the investigation of sequences and structures. *J Mol Biol 247*:536–540.

Needleman SB, Wunsch CD. 1971. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol 48*:443–453.

Orengo CA. 1994. Classification of protein folds. *Curr Opin Struct Biol 4*:429–440.

Orengo CA, Jones DT, Thornton JM. 1994. Protein superfamilies and domain superfolds. *Nature 372*:631–634.

Orengo CA, Swindells MB, Michie AD, Zvelebil MJ, Driscoll PC, Waterfield MD, Thornton JM. 1995. Structural similarity between the pleckstrin homology domain and verotoxin: The problem of measuring and evaluating structural similarity. *Protein Sci 4*:1977–1983.

Overington JP, Zhu ZY, Sali A, Johnson MS, Sowdhamini R, Louie GV, Blundell TL. 1993. Molecular recognition in protein families: A database of aligned three-dimensional structures of related proteins. *Biochem Soc Transact 3*:597–604.

Pearson WR. 1996. Effective protein sequence comparison. *Methods Enzymol 266*:227–259.

Pearson WR, Lipman DJ. 1988. Improved tools for biological sequence analysis. *Proc Natl Acad Sci USA 85*:2444–2448.

Perutz MF, Rossmann MG, Cullis AF, Muirhead H, Will G, North ACT. 1960. *Nature 185*:416–422.

Remington SJ, Matthews BW. 1980. A systematic approach to the comparison of protein structures. *J Mol Biol 140*:77–99.

Rossmann MG, Argos P. 1975. A comparison of the heme binding pocket in globins and cytochrome b5. *J Biol Chem 250*:7525–7532.

Rossmann MG, Liljas A, Branden CI, Banaszak LJ. 1975. *Enzymes 11*:61–102.

Russell RB, Barton GB. 1992. Multiple protein sequence alignment from tertiary structure comparisons. Assignment of global and residue level confidences. *Proteins 14*:309–323.

Sali A, Blundell TL. 1990. The definition of general topological equivalence in protein structures: A procedure involving comparison of properties and relationships through simulated annealing and dynamic programming. *J Mol Biol 212*:403–428.

Sali A, Overington JP. 1994. Derivation of rules for comparative protein modeling from a database of protein structure alignments. *Protein Sci 3*:1582–1596.

Sanchez R, Sali A. 1997. Advances in comparative protein modeling. *Curr Opin Struct Biol 7*:206–214.

Satow Y, Cohen GH, Padlan EA, Davies DR. 1987. Phosphocholine binding immunoglobulin Fab McPC603: An X-ray diffraction study at 2.7 Å. *J Mol Biol 190*:593–604.

Schuler GD, Epstein JA, Ohkawa H, Kans JA. 1996. Entrez: Molecular biology database and retrieval system. *Methods Enzymol 266*:141–162.

Smith RF, Smith TF. 1992. Pattern induced multi-sequence alignment (PIMA) algorithm employing secondary structure dependent gap penalties for use in comparative protein modelling. *Protein Eng 5*:35–41.

Sonnhammer EL, Eddy SR, Durbin R. 1997. Pfam: A comprehensive database of protein domain families based on seed alignments. *Proteins 28*:405-4-20.

Subbiah S, Laurents DV, Levitt M. 1993. Structural similarity of DNA-binding domains of bacteriophage repressors and the globin core. *Curr Biol 3*:141–148.

Taylor WR, Flores TP, Orengo C A. 1994. Multiple protein structure alignment. *Protein Sci 3*:2358–2365.

Taylor WR, Orengo CA. 1989. Protein structure alignment. *J Mol Biol 208*:1–22.

Thompson JD, Higgins DG, Gibson TJ. 1994. CLUSTAL W: Improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. *Nucleic Acid Res 22*:4673–4680.

Vingron M, Waterman MS. 1994. Sequence alignment and penalty choice. Review of concepts, case studies and implications. *J Mol Biol 235*:1–12.

Vogt G, Etzold T, Argos P. 1995. An assessment of amino acid exchange matrices in aligning protein sequences: The twilight zone revisited. *J Mol Biol 249*:816–831.

Vriend G, Sander C. 1991. Detection of common three-dimensional substructures in proteins. *Proteins 11*:52–58.

Zhu ZY, Sali A, Blundell TL. 1992. A variable gap penalty function and feature weights for protein 3-D structure comparisons. *Protein Eng 5*:43–51.