Definition of General Topological Equivalence in Protein Structures

A Procedure Involving Comparison of Properties and Relationships through Simulated Annealing and Dynamic Programming

Andrej Šali[†] and Tom L. Blundell[‡]

Laboratory of Molecular Biology
Department of Crystallography
Birkbeck College, University of London
London WC1E 7HX, England

(Received 26 June 1989; accepted 31 October 1989)

A protein is defined as an indexed string of elements at each level in the hierarchy of protein structure: sequence, secondary structure, super-secondary structure, etc. The elements, for example, residues or secondary structure segments such as helices or β-strands, are associated with a series of properties and can be involved in a number of relationships with other elements. Element-by-element dissimilarity matrices are then computed and used in the alignment procedure based on the sequence alignment algorithm of Needleman & Wunsch, expanded by the simulated annealing technique to take into account relationships as well as properties. The utility of this method for exploring the variability of various aspects of protein structure and for comparing distantly related proteins is demonstrated by multiple alignment of serine proteinases, aspartic proteinase lobes and globins.

1. Introduction

Protein evolution has led to families of structures that have the same "fold" at the level of super-secondary structures, motifs, domains or entire globular proteins (Richardson, 1977, 1981). The thermodynamically stable arrangements available to a polypeptide must satisfy a number of criteria, such as hydrophobicity and close packing of the protein interior and hydrogen bonding of peptide amide and carbonyl functions that become inaccessible to the aqueous environment. This is achieved mainly through α-helices and β-sheets, which can be close-packed in relatively few ways (Chothia, 1984). Thus, in evolution, tertiary structure is more conserved than the amino acid sequence and the number of stable folds is limited at each level in the hierarchical structure of proteins.

The consequences of this for protein modelling have long been recognized. Closely related homologous proteins of known three-dimensional structures can be used to model sequences of proteins of unknown tertiary structures (Browne *et al.*, 1969; Greer, 1981; for a review, see Blundell *et al.*, 1987). The basis of this modelling is the superposition of a number of three-dimensional structures of homologous proteins in order to define topological equivalence of amino acid residues (McLachlan, 1979, 1982; Schulz, 1980; KenKnight, 1984). This has been developed into a systematic approach in which several homologous structures can be used in modelling the unknown (Sutcliffe *et al.*, 1987*a*,*b*; Blundell *et al.*, 1988, Sali *et al.*, 1990). Thus, rules and procedures have been established to define the relative positions of the conserved secondary structural elements (i.e. the framework: Sutcliffe *et al.*, 1987*a*), to select appropriate fragments for the variable regions, which are often loops (Sibanda & Thornton, 1985; Chothia *et al.*, 1986; Sutcliffe, 1988; Sibanda *et al.*, 1989), and for the replacement of side-chains (Sutcliffe *et al.*, 1987*b*; Summers *et al.*, 1987; McGregor *et al.*, 1987; M. S. Johnson, unpublished results). In a parallel development, Jones & Thirup (1986) have shown that modelling using electron density during protein crystallography can also be aided by selection of fragments from a series of other proteins of known three-dimensional structure.

However, although proteins within families have the same tertiary fold, the secondary structural

[†] On leave from the Department of Biochemistry, J. Stefan Institute, Ljubljana, Yugoslavia.

[‡] Author to whom all correspondence should be sent.

0022-2836/90/060403-26 $03.00/0

elements may undergo deformations and relative translations and rotations to optimize packing of side-chains that have mutated during evolution (Lesk & Chothia, 1980). For certain divergent families of proteins, the root-mean-square deviation of superposed pairs has been shown to be related to the sequence differences (for example, see Wistow *et al.*, 1983). General relationships between sequence differences and root-mean-square differences of superposed, homologous proteins have been described by Chothia & Lesk (1986) and Hubbard & Blundell (1987). The picture is complicated by the fact that, for a family of homologous proteins, the divergence of sequences and the increase in the root-mean-square differences between equivalent atoms are accompanied by a decrease in the percentage of residues that can be considered to be equivalent by direct superposition. Thus, for two proteins with 30% identity in sequence, the topologically equivalent residues may have a root-mean-square difference of approximately 1·5 Å (1 Å = 0·1 nm) and comprise only 20 to 30% of the total number of residues (Hubbard & Blundell, 1987; Johnson *et al.*, 1989). This may provide an insufficient framework for modelling and emphasizes the requirement for more flexible procedures for defining topological equivalence.

The problem of defining topologically equivalent residues in polypeptides that adopt distantly similar folds was addressed more than a decade ago by Rossmann, Remington, Matthews and their collaborators (for a review, see Matthews & Rossman, 1985). In the first approach (Rao & Rossmann, 1973; Eventoff & Rossmann, 1975; Rossmann & Argos, 1975, 1976, 1977), the two structures are first least-squares fitted using the initial set of equivalent residues. The equivalences are then updated according to both the distances between potentially equivalent C^{α} atoms and local directions of the main chain. The superposition and updating is repeated until no increase in a number of equivalences can be achieved. Alternatively, if no initial set of equivalent residues can be obtained, an exhaustive search for an initial superposition is made by systematically varying the three Eulerian angles that determine the relative orientation of the two structures.

In the second approach (Remington & Matthews, 1978, 1980), windows consisting of the conformation of n contiguous residues in the first structure are defined, and then compared locally by least-squares fitting to every part of the second structure. This gives a difference matrix representing the scores for each of the $(N-n+1) \times (M-n+1)$ pairs of segments where the two proteins have N and M residues. The difference matrix can then be used to infer the alignment of the two structures.

Both procedures allowed distantly related structures such as nucleotide binding proteins (Eventoff & Rossmann, 1975; Rossmann & Argos, 1976, 1977), cytochromes and globins (Rossmann & Argos, 1975, 1976, 1977; Argos & Rossmann, 1979), mammalian and microbial serine proteinases (Remington & Matthews, 1980) and vertebrate and phage lysozymes (Rossmann & Argos, 1976, 1977; Remington & Matthews, 1978, 1980; Matthews *et al.*, 1981) to be systematically compared.

An alternative approach, in which secondary structural elements, represented by vectors, are compared, has been suggested by Murthy (1984). This procedure provides an attractive simplification to protein comparison as elements of secondary structure (α-helices and β-strands) are rarely interchanged in folds of similar topology. The approach has been used by Richards & Kundrot (1988), who systematically compared local relationships between secondary structural elements in their database search for a given secondary structure pattern.

In fact, it may be advantageous for protein comparisons to operate simultaneously at several levels in the hierarchy of the protein structure. The multilevel representation of protein organization has been exploited by Lathrop *et al.* (1987), who used artificial intelligence procedures to find a predefined hierarchical pattern in a given protein sequence.

In our approach to protein comparison, we first define the protein as an indexed string of elements that may exist at several levels in the protein hierarchical organization: residue, secondary structure, supersecondary structure, motif, domain or globular structure. We then associate with every element features that indicate a common fold. At each level, the features compared could be properties or relationships. Residue properties include sequence identity, hydrophobicity, size of side-chain, charge, etc. When three-dimensional structures are used in the comparison, the residue properties might also include local conformations, the orientation of side-chains and main chains compared to the centre of mass of the globular structure, accessibilities, ϕ and ψ dihedral angles, positions in space and local main-chain directions in the least-squares-fitted molecules. Equivalent properties concerning higher levels of structure can also be aligned. These include the nature of the secondary structure element i, its accessibility, the orientation of the vector defining the helix or strand compared to the centre of mass and the improper dihedral angle formed by secondary structure elements $i-1, i, i+1$. Comparison of all such properties can be incorporated in a residue-by-residue weight matrix of $N \times M$ elements where N and M are the numbers of residues in the two proteins compared. In this paper we describe such weight matrices and show how the optimal alignment can then be derived using the dynamic programming approach of Needleman & Wunsch (1970).

We also discuss the use of more powerful features in comparisons at each level of the hierarchical structure of proteins. We show how specific relationships such as hydrogen bonding interactions or packing relations, which tend to be conserved in protein folds, can be used in alignment procedures. However, a relationship affects more than one element in a sequence and this makes the conven-

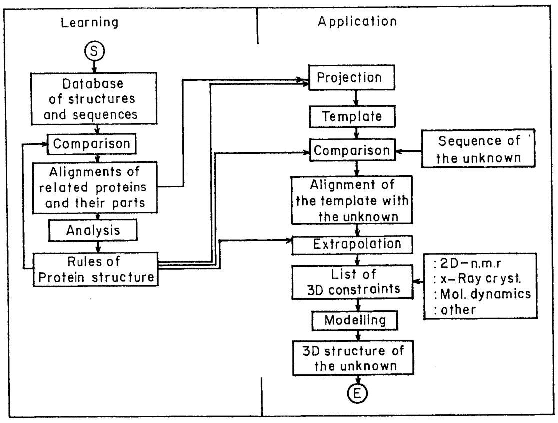


Figure 1. Knowledge-based modelling of proteins. In the learning part, one starts with the comparison of proteins and fragments from databases of sequences and structures. The results of this operation are the alignments of related proteins and their parts. In the 2nd step, these alignments are then analysed and some knowledge about protein structure is gained. This knowlege includes rules that are later employed to improve the quality of the comparison of proteins, which can be part of the derivation of templates and which can also be used in obtaining 3-dimensional (3D) constraints on the sequence of the unknown. The application part of the knowledge-based approach to protein modelling starts with the construction of the template for the assumed fold of the unknown. The template is, in fact, a generalized protein defined in the same way as we defined the protein for the purpose of comparison: a template is an indexed string of elements that are associated with a number of features. The difference between the template and the protein is that the template summarizes the knowledge about *all* known proteins conforming to the selected fold. In the 2nd step, the template is compared with the sequence of the unknown and the alignment between the 2 is obtained. In the 3rd step, this alignment and the rules of protein structure are used to derive a list of spatial constraints on the 3-dimensional structure of the unknown. This list will include, for example, conserved hydrogen bonds, invariable secondary structure, constraints from application of the rules of protein structure on the sequence of the unknown, etc. In the 4th and last step, the structure of the unknown is obtained by the minimization of violations of the constraints from the previous step. Procedures such as those developed for constructing protein models from bounds on distances obtained from 2D n.m.r. (2-dimensional nuclear magnetic resonance) experiments (Crippen, 1977; Havel *et al.*, 1983; Braun & Go, 1985) can be applied. Due to the nature of the last modelling step, information from a variety of both experimental and theoretical sources can be used in addition to the knowledge from the known structures. The areas that, in principle, could contribute to this unified scheme of protein modelling include 2D-n.m.r., X-ray crystallography and molecular dynamics.

tional dynamic programming approach inapplicable. Instead, we describe how simulated annealing optimization can be applied to provide an initial set of equivalences based on relationships, and these can then be introduced directly into the residue-by-residue weight matrix.

Our main purpose for developing a new protein comparison method is to apply it in the modelling of proteins by homology and analogy. Knowledge-based protein modelling can be defined as a prediction of protein three-dimensional structure from the amino acid sequence where the main source of information comes from the exploration of the known structures. In Figure 1, we propose a flow chart for such an effort. The whole scheme can be divided into two parts. The first part, *learning*, is concerned with the acquisition of a general knowledge about protein architecture, predominantly in the form of rules. This is achieved by the comparison of known proteins and their parts, both at the sequence and structural level. The second part of the scheme, *application*, describes how this knowledge is used to model a particular protein. Again, comparison of a sequence of an unknown with known homologous structures is an important part of the structure prediction.

The importance of the comparison approach described here is that it demands as well as initiates new methodologies for various steps of knowledge-based protein modelling. In fact, it was the definition of the protein as a string of elements having certain properties and being engaged in certain relationships that guided the generalization of the modelling by homology described above. Procedures for each of the steps in Figure 1 will be described in later papers.

2. Methods

(a) *General definitions*

We define a protein as an hierarchy of structures: sequence, secondary structure, supersecondary structure, motif, domain or entire protein. At each level in the hierarchy we define a string of indexed elements (amino acid residues, secondary structural segments, etc.) each of which is associated with a series of properties and engaged in a number of relationships with other elements at the same level. Table 1 lists some of these features at the first 2 levels in the hierarchy of protein structure.

We define a normalized difference, ${}^n w^f_{ij}$, in a certain feature f between the residues i and j from the first and second protein, respectively. We also define a scaling factor ρ^f, which determines a relative importance of feature f used in the comparison. This then allows a weighted sum W_{ij} to be calculated as:

$$W_{ij} = \sum_{l} \left(\sum_{p} \rho^{p}\, {}^{n}w^{p}_{ij} + \sum_{r} \rho^{r}\, {}^{n}w^{r}_{ij} \right). \qquad (1)$$

This summation runs over all levels of the hierarchy of the structure (indicated by an index l). At each level, there are terms for a number of features that are conveniently classified into properties (superscript p) and relations (superscript r). It is this residue-by-residue weight matrix $\mathscr{W}$ consisting of elements W_{ij} that is used in the dynamic programming procedure described by Needleman & Wunsch (1970) to align the 2 proteins that are being compared.

The original weights for individual features, w^f_{ij}, are defined in eqns (4) to (18). However, before applying

Table 1

Some of the features that can be used in the comparison of protein structures

Residues	Segments
Properties	
Identity	Secondary structure type
Residue type properties	Amphipathicity
Local conformation	Improper dihedral angle
Distance from gravity centre	Distance from gravity centre
Side-chain orientation	Orientation relative to gravity centre
Main-chain orientation	
Solvent accessibility	Solvent accessibility
Position in space	Position in space
	Orientation in space
Relations	
Hydrogen bond	
Distances to 1 or more nearest neighbours	Distances to 1 or more nearest neighbours
Disulphide bond	Relative orientation of 2 or more segments
Ionic bond	
Hydrophobic cluster	

Various features are represented by rows and different levels of protein organization by columns. Only residue and secondary structure levels are shown here. The term property is used for all protein features that imply comparison of only 1 element from each protein. Conversely, the term relationship is used for features that imply explicit comparison of at least 2 elements from each protein.

Table 2

Features and some parameters used for the alignments in this paper

Index	Brief description	w^f	d^f_c
1	Residue local fold	1·00	3·0 Å
2	Residue type properties	1·00	2·0
3	Residue distance from MGC	0·40	6·0 Å
4	Side-chain orientation relative to MGC	0·06	145°
5	Side-chain orientation relative to main-chain	0·03	145°
6	Main-chain orientation relative to MGC	0·06	145°
7	Side-chain solvent accessibility	0·025	100%
10	Main-chain solvent accessibility	1·00	100%
14	Hydrogen bonding relationship	1·00	1·0
15	Residue identity	1·00	1·0
17	Residue position in space	0·01	2·5 Å
18	ϕ dihedral angle	0·01	145°
19	ψ dihedral angle	0·01	145°
20	Main-chain directions	0·10	9·0 Å

For the definitions of the features and parameters w^f and d^f_c, see Methods. MGC, molecular gravity centre.

eqn (1), these weights are normalized using the following common transformation:

$$ {}^{n}w^f_{ij} = \omega^f \begin{cases} w^f_{ij} & \text{if } w^f_{ij} \le d^f_c \\ d^f_c & \text{if } w^f_{ij} > d^f_c . \end{cases} \qquad (2) $$

Parameter d^f_c is a cutoff constant that is usually set to a random value of original weights w^f_{ij} for feature f. This transformation of original residue-by-residue weights was introduced because it is convenient to know the upper limit on the weight matrix elements W_{ij} when choosing gap penalties (section (d)(i), below) and when calculating protein–protein distance scores (section (f), below). An additional parameter is a scaling factor ω^f whose only function is to bring the differences in all features to the same order of magnitude. For example, the difference in angles ranges between 0 and 2π *radians*, whereas the difference in the distances from the molecular centres of gravity can be as big as a few 10s of ångstrom units. Hence, scaling using ω^f is advantageous because it allows easy application of weights ρ^f in eqn (1) specifically for adjusting the relative importance of individual features. The parameters associated with the features used for the alignments in this paper are summarized in Table 2. A particular choice of cutoff constants d^f_c has, in general, a negligible influence on the alignment compared to the feature scaling factors ρ^f, and was therefore not optimized to obtain better results in this paper.

When calculating residue-by-residue weights w^f_{ij} based on features f of elements at levels higher than a residue level, the following common equation is used:

$$ w^f_{ij} = \begin{cases} w^{lf}_{i'j'} & \text{if residues } i \text{ and } j \text{ belong to segments } i' \text{ and } j' \\ d^f_c & \text{if residues } i \text{ and } j \text{ are not contained in any segment.} \end{cases} \qquad (3) $$

Weight $w^{lf}_{i'j'}$, describes a difference in the feature f between the two segments i' and j', at the level l, that include the ith and jth residue from the first and second protein, respectively.

Using both sequence and structural information, we now proceed to define individual weights w^f_{ij} at the residue (sections (b)(i) to (b)(xiii), below) and secondary structure levels (section (b)(xiv), below). We distinguish between properties (section (b), below), which are features associated with one element only, and relationships (section (c), below), which are features associated with more than one element. Then we describe a new simulated annealing algorithm that aligns sequences of relationships (section (c)(ii), below) and we consider our implementation of the dynamic programming procedure of Needleman & Wunsch (1970) (section (d)(i), below). Finally, we extend the pairwise structural alignment to simultaneous comparison of several proteins (section (e), below) and show how to use the information from the multiple alignment to classify proteins (section (f), below).

(b) *Properties*

(i) *Residue identity*

The term w^{15}_{ij} is based on sequence information alone:

$$ w^f_{ij} = \begin{cases} 0 & \text{if residues } i \text{ and } j \text{ are identical} \\ 1 & \text{if residues } i \text{ and } j \text{ are not identical.} \end{cases} \qquad (4) $$

Using only this feature in the definition of the residue-by-residue weight matrix $\mathscr{W}$ (eqn (1)) would be analogous to early sequence alignment methods that distinguish only the residue identities (for example, see Needleman & Wunsch, 1970).

(ii) *Residue type properties*

This term is based on the analysis by Argos (1987) of residue properties for sequence alignment. He obtained a combination of the 5 physico-chemical and statistical parameters that gave the best results. These properties include the surrounding hydrophobicity (Manavalan &

Ponnuswamy, 1978), turn preference (Palau *et al.*, 1982), residue bulk (Jones, 1975), residue refractivity index (Jones, 1975) and antiparallel strand preference (Lifson & Sander, 1979). They are presumably the most conserved residue characteristics in evolution (Argos, 1987). We define a residue type's properties term $w^2_{r_i,r'_j}$ as a scaled Euclidean distance between residue types i and j in a 5-dimensional space of the 5 normalized properties:

$$w^2_{r_i,r'_j} = \sqrt{\sum_{p=1}^{5} (x_{ip} - x_{jp})^2}, \qquad (5)$$

where a value for the normalized property p of a residue type i, x_{ip}, is calculated from the original values x'_{ip} as follows:

$$\begin{aligned} x_{ip} &= (x'_{ip} - \bar{x}'_p)/\sigma_p && p = 1, 2, \ldots, 5 \\ \sigma_p &= \sqrt{\frac{\sum_i^N (x'_{ip} - \bar{x}'_p)^2}{N}} && p = 1, 2, \ldots, 5, \quad N = 20 \qquad (6) \\ \bar{x}'_p &= \frac{1}{N} \sum_{i=1}^N x'_{ip} && p = 1, 2, \ldots, 5, \quad N = 20. \end{aligned}$$

Indices i and j code for the 2 amino acid residue types compared. Indices r_i and r'_j are residue numbers in the 1st and 2nd sequence, respectively. Twenty components of $\bar{x}_p$ are normalized to the distribution with a mean of 0 and a root-mean-square deviation of 1.

The feature based on residue type properties was defined to distinguish between residues in a more refined way than does the sequence identity. A familiar example of an equivalent sequence alignment technique is an ALIGN program by Dayhoff *et al.* (1983) that employs the MDM_{250} similarity matrix. In this paper, it was more convenient to define a distance measure from scratch than to transform the accepted MDM_{250} similarity matrix into the distance matrix. In any case, the cross-correlation coefficient between the 2 is $-0{\cdot}65$.

(iii) *Residue local fold*

This term describes the difference in local conformation of 2 short chain segments, 1 from each of the proteins compared. Segments are defined by a small number of contiguous C^α atoms on each side of the central C^α atom. The local fold term then applies to these central residues. The difference w^1_{ij} is defined on the basis of intra-segment atomic distances:

$$w^1_{ij} = \frac{1}{N} \sum_{\substack{l,k=-(\alpha-1)/2 \\ k>l \\ k-l\geq\beta \\ k-l\leq\gamma}}^{(\alpha-1)/2} |C_{i+l,i+k} - C'_{j+l,j+k}|. \qquad (7)$$

$\mathscr{C}$ is a protein distance matrix derived from positions of C^α atoms only. Parameter α is the length of a segment, which should be at least 5 and odd. Parameter β restricts the sum only to the distances defined by C^α atoms that are at least β positions apart. Parameter γ restricts the sum to distances between C^α atoms that are at most γ residues apart (γ must be smaller than α). The usual values for parameters α, β and γ are 7, 2 and 6, respectively. N is the number of terms in the summation.

(iv) *Residue distance from molecular centre of gravity*

The centre of gravity of the molecule is defined by positional vectors $\mathbf{X}_i$ of N C^α atoms:

$$\mathbf{O} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{X}_i.$$

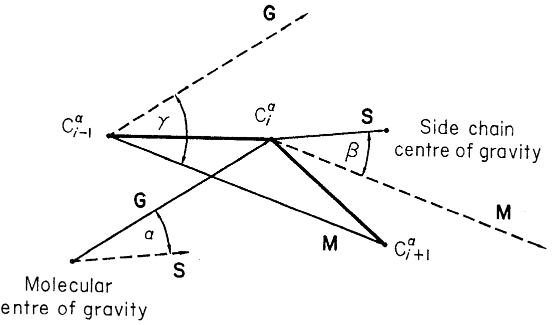


Figure 2. Definition of orientation angles for main chain and side-chain. Angle α describes the orientation of the side-chain relative to the molecular centre of gravity, angle β quantifies the orientation of the side-chain relative to the main chain, and angle γ describes the orientation of the main chain relative to the molecular centre of gravity. This definition of orientation angles allows comparison of 2 molecules that are not necessarily superposed.

The vector from the molecular centre of gravity to the ith C^α atom is: $\mathbf{R}_i = \mathbf{X}_i - \mathbf{O}$. The term w^3_{ij} describing the difference in the distances from the centres of gravity is then:

$$w^3_{ij} = ||\mathbf{R}_i| - |\mathbf{R}'_j||. \qquad (8)$$

(v) *Orientation of the side-chain relative to the molecular centre of gravity*

Side-chain direction, $\mathbf{S}$, is defined by the vector from a C^α atom to the side-chain centre of gravity. This is computed as a non-weighted mean of positions of all non-hydrogen side-chain atoms, excluding the C^α atom. The orientation of the side-chain relative to the molecular centre of gravity can then be defined as the angle, α, between 2 vectors, the 1st being the vector from the molecular centre of gravity to the C^α atom, $\mathbf{G}$, and the 2nd being the absolute side-chain direction $\mathbf{S}$ (Fig. 2): $\cos\alpha = (\mathbf{G}\cdot\mathbf{S})/(GS)$. The term w^4_{ij} then describes the difference between the 2 orientations of the side-chain relative to the molecular centre of gravity:

$$w^4_{ij} = |\alpha_i - \alpha'_j|. \qquad (9)$$

Weight w^4_{ij} is set to 0 when at least one of the residues compared is Gly. The precise choice of this "zero" constant has a minimal effect on the final alignment: an equivalence of Gly in one protein to any of the residues in the other protein has the same score whatever the constant and therefore no particular alignment is favoured. If the constant is increased, the alignment can be affected only as a result of the gap at Gly being preferred but this was found to occur very rarely and the simplest choice of zero was adopted.

(vi) *Orientation of the side-chain relative to the main chain*

The main-chain direction at the C^α_i atom, $\mathbf{M}$, is defined by the vector from the C^α_{i-1} to the C^α_{i+1} atom. The orientation of the side-chain relative to main-chain can then be defined as the angle, β, between vectors $\mathbf{M}$ and $\mathbf{S}$ (Fig. 2). The term w^5_{ij} that describes the difference between the 2 orientations of the side-chain relative to main chain is:

$$w^5_{ij} = |\beta_i - \beta'_j|. \qquad (10)$$

Weight w_{ij}^{5} is set to 0 when at least one of the residues compared is Gly.

(vii) *Orientation of the main chain relative to the molecular centre of gravity*

The orientation of the main chain relative to the molecular centre of gravity can be defined as the angle, γ, between vectors **G** and **M** (Fig. 2). The term w_{ij}^{6} then describes the difference between the 2 orientations of the main chain relative to the molecular centre of gravity:

$$w_{ij}^{6} = |\gamma_i - \gamma_j'|. \tag{11}$$

(viii) *Side-chain accessibility*

The solvent contact areas for all atoms in a protein were calculated using the algorithm of Richmond & Richards (1978). Whether or not all ligands, domains and subunits are included in these calculations depends on the particular case. For example, comparison of the lobes of aspartic proteinases does not improve with the addition of the other lobe in the accessibility calculation. On the other hand, the alignment of the nucleotide binding domains of dehydrogenases might be improved by the inclusion of the nucleotides when accessibilities are calculated. The solvent radius was 1·4 Å. Side-chain accessibility calculations and normalization were carried out as described by Hubbard & Blundell (1987). The normalized side-chain accessibilities, a_i^s, which are comparable between different residue types, were then used to define the difference in side-chain accessibilities:

$$w_{ij}^{7} = |a_i^s - a_j'^s|. \tag{12}$$

(ix) *Residue main-chain accessibility*

The definition of the difference in residue main-chain accessibilities, w_{ij}^{10}, is based on that described above for a difference in side-chain accessibilities, except that main-chain accessibility, a_i^m, is substituted for side-chain accessibility:

$$w_{ij}^{10} = |a_i^m - a_j'^m|. \tag{13}$$

(x) *Main-chain dihedral angle ϕ*

Given the dihedral angles ϕ_i and ϕ_j' for the ith and jth residue from the 1st and 2nd protein, respectively, the difference in the main-chain dihedral angles ϕ is defined as:

$$w_{ij}^{18} = \begin{cases} |\phi_i - \phi_j'| & \text{if } (\phi_i - \phi_j'| \leq 180^\circ \\ 360^\circ - |\phi_i - \phi_j'| & \text{if } |\phi_i - \phi_j'| > 180^\circ. \end{cases} \tag{14}$$

(xi) *Main-chain dihedral angle ψ*

The term w_{ij}^{19} for the ψ dihedral angle is obtained using an equation analogous to eqn (14) for the ϕ dihedral angle:

$$w_{ij}^{19} = \begin{cases} (\psi_i - \psi_j'| & \text{if } |\psi_i - \psi_j'| \leq 180^\circ \\ 360^\circ - |\psi_i - \psi_j'| & \text{if } (\psi_i - \psi_j'| > 180^\circ. \end{cases} \tag{15}$$

(xii) *Absolute position in space*

The 2 protein structures are first superposed by the program MNYFIT (Sutcliffe *et al.*, 1987*a*), which implements the least-squares superposition procedure of McLachlan (1982). The weight reflecting the difference in the position of residues i and j from the 1st and 2nd protein, respectively, is then defined as:

$$w_{ij}^{17} = |\mathbf{R}_i - \mathbf{R}_j'|, \tag{16}$$

where $\mathbf{R}_i$ and $\mathbf{R}_j$ are positional vectors of the ith and jth C^α atoms from the 1st and 2nd protein, respectively. Using this feature enables the incorporation of information from the least-squares-fitting approach to protein comparison into our alignment method.

(xiii) *Main-chain directions*

The difference in directions of the vectors spanning C^α atoms within 2 segments that are centred at the ith and jth residue of the 1st and 2nd proteins, respectively, is calculated using the co-ordinates of least-squares-fitted structures. The weight is:

$$w_{ij}^{20} = t_x + t_y + t_z, \tag{17}$$

where each of the t terms is calculated using eqn (7), wherein the matrices of corresponding orthogonal components of the intramolecular distances are used instead of the intramolecular distance matrix $\mathscr{C}$.

(xiv) *Properties of secondary structure elements*

At the 2nd level in the hierarchy, we define elements roughly as secondary structural units. For this, we applied the following steps.

(1) Use the program DSSP for secondary structure definition (Kabsch & Sander, 1983) to classify every residue into 1 of the 4 types: helical (DSSP residue codes I, H or G), β-strand (codes E and B), bend (codes S and T) and extended (in DSSP coded by a blank).

(2) Define helical segments as all homogeneous stretches where the number of consecutive residues of the helical type is at least N_α (usually 4). Define β-strands as all stretches between bend or helical residues that contain at least N_β (usually 3) residues of the β-strand type. Define segments of the extended type as the remaining stretches of at least N_ρ (usually 4) residues that do not contain any bend or helical residue.

(3) Test every segment that is at least as long as twice its minimal length (see step (2)) for a quality of the least-squares line through its C^α atoms: select several positions in the middle of a segment as a break to obtain 2 halves; fit 2 least-squares lines to the C^α atoms of each half; if the angle or the distance between the 2 least-squares lines for any of the breaks is outside the allowed range (if smaller then 150° for an angle, or greater then 3 Å for a distance) then the division of the segment into 2 segments is retained; otherwise, the least-squares line through all C^α atoms, where projections of the terminal C^α atoms determine the endpoints, is taken as a fair representative of the segment's position and orientation.

(4) Inspect the C^α backbone and segments on a graphics terminal and make appropriate changes to the division if necessary. Most likely candidates are helices that are frequently too short in the DSSP definition. Recalculate least-squares lines and endpoints for the new division.

In general, one treats the secondary structure elements in a similar way as the lower level elements, the amino acid residues. The properties include the type of the secondary structure of the segment (helical, β-strand or extended), improper dihedral angle of the segment defined by segments $i-1$, i and $i+1$, the average normalized side-chain and main-chain accessibilities of the residues in the segment, the distance and orientation of the segment with respect to the molecular centre of gravity and position and orientation of the segment when a molecule is in a reference orientation obtained from the rigid-body superposition of the structures compared.

(c) *Relationships*

(i) *Definitions of relationships*

Relationships between residues, secondary structural elements or higher-order elements of the protein structure can be completely specified by several attributes. These define the type of a relation, e.g. covalent or hydrogen bond, and the type of elements, e.g. donor or acceptor of a hydrogen bond. A further attribute defines the multiplicity, e.g. a hydrogen bond is a binary relationship but electrostatic and hydrophobic interactions might be multiple. However, it is convenient to consider such multiple relationships as a number of binary relations, since this renders all relationships susceptible to a single mathematical treatment. Finally, we define the strength of the relation, e.g. the distance to a neighbour or the energy of a hydrogen bond.

In most cases the attributes are easy to calculate. For main-chain–main-chain hydrogen bonds, we used the program DSSP (Kabsch & Sander, 1983) to generate hydrogen atoms and compute the electrostatic energy of the hydrogen bond. Every pair of a carbonyl oxygen atom and an amide nitrogen atom with an energy less than E_c (usually $-1{\cdot}0$ kcal/mol; 1 cal $= 4{\cdot}184$ J) was then defined as a hydrogen bond. When bifurcated bonds were obtained, only the one with the most favourable energy was retained.

For hydrophobic relationships, we first flagged the "hydrophobic" residues. Hydrophobic residues are defined here as all residues of the type A, T, V, M, I, L, F, W, H, C, K and Y whose fractional side-chain solvent contact area is at most S_h (usually 20%). The potentially hydrophobic residue types were obtained from the Venn diagram classification of amino acid residues (Fig. 3 of Taylor, 1986). It may be noted that the inclusion of residues such as histidine and lysine does not preclude their classification as polar residues; this is a consequence of the ambivalent nature of some of the residues with regard to hydrophobicity. Additionally, not all leucine, valine residues, etc., are "hydrophobic" according to this definition, since a hydrophobic residue has to be buried too. We then defined the pair of residues in a hydrophobic relationship as every 2 hydrophobic residues that have at least N_h contacts (usually 1) between side-chain atoms at a distance less than d_h (usually 4·5 Å). The hydrophobic residues next to each other in a sequence were treated slightly differently in that the side-chain C^β atoms were not taken into account.

(ii) *Combinatorial simulated annealing technique for comparison of relationships*

Although relationships such as hydrogen bonds are easy to define, the identification of topological equivalence based on the relationships is more problematical, as each relationship involves at least 2 element-by-element matches. For example, for a hydrogen bond between residues a_1 and a_2 of protein A to be equivalent to a hydrogen bond between b_1 and b_2 of protein B, residues a_1 and b_1 as well as a_2 and b_2 must be equivalent.

In general, it is beyond the dynamic programming algorithm of Needleman & Wunsch (1970) (section (d)(i), below) to match relationships (for an explanation see the legend to Fig. 3). Nevertheless, the dynamic programming approach can be generalized for this alignment (Fig. 3). However, contrary to the Needleman & Wunsch (1970) case, fast evaluation of the dynamic programming formula by $N \times N$ iteration (Needleman & Wunsch, 1970; Sankoff & Kruskal, 1983) is not possible for the generalized formula. The reason is the dependence of weights for an alignment of relationships on the already assigned equivalences. As a result, the generalized formula has to be evaluated by a straightforward but slow combinatorially explosive recursion; computer time for evaluation of such a generalized formula rises slightly faster than exponentially with sequence length and is already impractical for the sequences of only 20 hydrogen bonds. To overcome this computational problem, we apply a simulated annealing minimization to obtain an alignment of 2 structures on the basis of relationships alone. We then use this information in the calculation of residue-by-residue weight matrix elements w^r_{ij} that contribute to the overall weights W_{ij} defined in eqn (1).

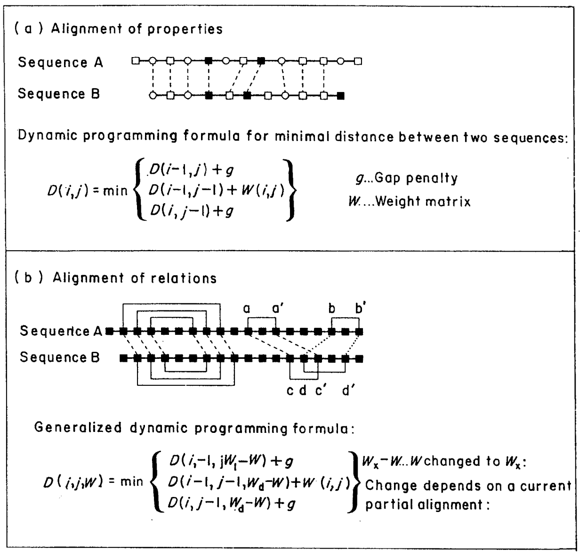

(a) Alignment of properties
Sequence A
Sequence B
Dynamic programming formula for minimal distance between two sequences:
D(i,j) = min { D(i−1,j) + g ; D(i−1,j−1) + W(i,j) ; D(i,j−1)+g }
g...Gap penalty
W....Weight matrix
(b) Alignment of relations
a a' b b'
Sequence A
Sequence B
c d c' d'
Generalized dynamic programming formula:
D(i,j,W) = min { D(i,−1, jW1−W) + g ; D(i−1, j−1, Wd−W)+W (i,j) ; D(i, j−1, Wd−W) + g }
Wx−W...W changed to Wx:
Change depends on a current
partial alignment:

Figure 3. Dynamic programming algorithms for alignment of sequences. (a) A sketch of the Needleman & Wunsch (1970) dynamic programming algorithm that takes into account residue properties to obtain the most parsimonious alignment of 2 sequences. Different geometrical symbols in the sequence stand for different residue types. Broken lines indicate equivalences between residues from the 2 sequences. (b) A generalization of the dynamic programming algorithm that can include relationships as well as properties into the comparison of the 2 sequences. Relationships, for example hydrogen bonds, are indicated by thin lines connecting 2 residues within 1 sequence. To see why the generalization is necessary for incorporation of relationships, consider this example. Suppose that during the recursive evaluation of a dynamic programming formula we had already equivalenced residues *a* and *c* and we were about to make the optimal selection between the following 3 alternatives: (1) residue *b* is an insertion, (2) residues *b* and *d* are equivalent and (3) residue *d* is an insertion. It is obvious from the Figure that the match between residues *a′* and *c′*, which is *only implied* by the already assigned match between *a* and *c*, must be considered during this selection, even if residue *c′* comes after residue *d* in the sequence *B*. In this example, the implied equivalence between *a′* and *c′* prevents the match between *b* and *d*. These considerations are sketched in the generalized formula where the residue-by-residue weights in the matrix $\mathcal{W}$ are treated as variables that depend on the current partial alignment.

In fact, several alignments of relationships for the same pair of structures are obtained, since the simulated annealing minimization does not guarantee the global minima. This then allows a calculation of an element-by-element weight matrix $\mathscr{U}^r$ with scalars U^r_{ij} that count how many times elements i and j from the 1st and 2nd protein, respectively, were matched. Superscript r stands for the types of relationship used in the alignment. The weight w^r_{ij} for relationship r is then given by:

$$w^r_{ij} = \frac{U^r_{max} - U^r_{i,j}}{U^r_{max}}, \tag{18}$$

where U^r_{max} is the largest element of matrix $\mathscr{U}^r$.

Simulated annealing was first introduced by Metropolis *et al.* (1953) for simulation of thermodynamic systems. A useful introduction may be found in Kirkpatrick *et al.* (1983).

A simulated annealing algorithm for an alignment of 2 sequences of relationships will now be described in detail. We begin by simplifying the sequence of elements engaged in any relations into a sequence of "bits", where every bit participates in precisely one binary relation. Four attributes are then needed to describe each binary relation in this sequence: the type of relation, the type of bits, the strength of relation and the original element indices of bits. The simplification is carried out in several stages (Fig. 4). First, each element is split into a number of bits: exactly one bit for every underlying binary relation of this element. This also means that every element that is not a member of any relation is omitted from the sequence of bits. Second, every bit is assigned a binary relation (Step 2 in Fig. 4). Observe that the notation of bits still keeps track of the element origin of each bit. Third, the sequence of bits from the 2nd step is transformed into the sequence of binary relations. This means that the numbering scheme indexes relations and not bits, the index is increased every time the first of the bits in a binary relation is passed over when going from the first to the last bit in a sequence (Step 3 in Fig. 4). It is these simplified sequences of binary relations, not the sequences of elements, which are actually compared in a simulated annealing algorithm.

In order to use the combinatorial simulated annealing technique we must specify 4 parts of its framework (Figs 5 to 7). The 1st part is the configuration space (Fig. 5), which is defined by all possible alignments of the 2 sequences of binary relations. Every alignment is symbolized by 1 point in the configuration space and is conveniently represented by a configuration vector. It should be noted that, if the bits i and j from the 1st bit sequence form the kth binary relation, and if the bits m and n from the 2nd bit sequence form the lth binary relation, and if one matches bits i and m, then bits j and n are automatically equivalenced too. If the 2 sequences have N_a and N_b binary relations (i.e. they are $2 \times N_a$ and $2 \times N_b$ bits long), then every alignment can be represented by the N_a dimensional vector $\mathbf{V}$. The ith component of the configuration vector, V_i, specifies the index of the relation in the 2nd sequence that is equivalent to the ith relation in the 1st sequence. If V_i is 0, then the ith binary relation and its 2 bits from the 1st sequence have no equivalents in the 2nd sequence. There are only 2 constraints in our definition of the configuration vector. First, there can be any number of zeros in $\mathbf{V}$, but every positive integer must occur at most once, i.e. we allow only for 1:1 equivalences. Second, the positive integers V_i must rise with an index i, i.e. relations have a sequential order.

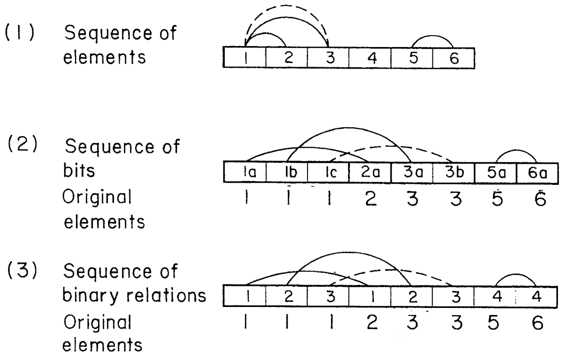


Figure 4. The 1st step in simulated annealing alignment of 2 sequences of relationships: the simplification of the sequence of elements, into the sequence of bits. The distinction between an element and a bit is that an element can be engaged in any number of any type of relationships, whereas a bit is only involved in one binary relationship. The continuous and broken lines connecting elements and bits represent 2 types of relationship. For further description see Methods, section (c)(ii).

The 2nd part of the combinatorial simulated annealing framework includes the moves in the configuration space (Fig. 5). The moves change the configuration vector from one allowed state to another allowed state. We define 3 kinds of move. The 1st is an insertion. This is achieved by substituting null V_i with a positive integer larger than its left, and smaller than its right, non-zero neighbour, respectively. The 2nd move is a deletion. This is achieved by substituting the non-null V_i with a 0. The 3rd move is a combination of a deletion and an insertion in this order.

The 3rd part of a simulated annealing is a function to be minimized. In general, the most parsimonious comparison can be that which minimizes the following

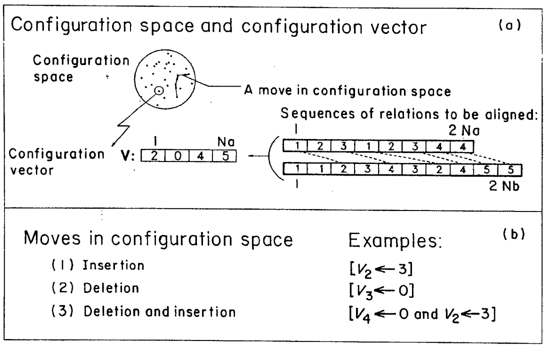

Configuration space and configuration vector
(a)
Configuration space
A move in configuration space
Sequences of relations to be aligned:
1
2 Na
1 2 3 1 2 3 4 4
1 1 2 3 4 3 2 4 5 5
1
2 Nb
Configuration vector
V:
1
Na
2 0 4 5
Moves in configuration space
(b)
(1) Insertion
(2) Deletion
(3) Deletion and insertion
Examples:
[$V_2 \leftarrow 3$]
[$V_3 \leftarrow 0$]
[$V_4 \leftarrow 0$ and $V_2 \leftarrow 3$]

Figure 5. Definition of a configuration space, a configuration vector and moves in the configuration space in the simulated annealing procedure. (a) A configuration vector describes the alignment of 2 sequences of relationships. A configuration space is defined by all possible states of the configuration vector (dots in the circle represent all alignments). A sequence of relationships is obtained from the last step in Fig. 4. The broken lines between the 2 sequences of relationships indicate the equivalences. (b) Definition of moves in a configuration space that change the state of a configuration vector and thereby the alignment as well. An assignment of j to V_i (i.e. an insertion) can occur if V_i is 0 and the left and right non-zero neighbours of V_i are smaller and larger than j, respectively. An assignment of 0 to V_i (i.e. a deletion) can occur if V_i is larger than 0.

function P:

$$P = \sum_{i=1}^{N_a} \begin{cases} R_{i,V_i} & \text{if } V_i > 0 \\ 1 & \text{if } V_i = 0, \end{cases} \tag{19}$$

where N_a is a number of binary relations in the 1st bit sequence and R_{ij} is an element of a distance matrix $\mathscr{R}$ which is proportional to the difference between the ith and jth binary relation of the 1st and 2nd bit sequence, respectively. R_{ij} is defined on the basis of the binary relation type, bit types and strength attribute and generally assumes values between 0 for identical relationships and 1 for dissimilar relationships. A constant 1 has a role of a gap-penalty and is arbitrary as long as it is in a context of the R_{ij} scale. R_{ij} can be larger than 1 if the equivalence of the relationships i and j is to be avoided more than is a gap.

Note that the definition of the configuration space and the moves is extremely simple. However, the consequence is that it allows for alignments of binary relations that are not meaningful (Fig. 6). There are 2 reasons for this. To describe the 1st we have to introduce the "equivalence line" term. The equivalence line is a line connecting 2 equivalent bits from the 1st and 2nd bit sequence, respectively. Every binary relation equivalence is represented by 2 equivalence lines. Imagine now the alignment as a set of all equivalence lines between the 2 sequences, the 2nd one aligned below the 1st. Then the 1st drawback of the simple configuration vector definition will be manifested in a possibility for crossings of the equivalence lines (Fig. 6). The 2nd drawback originates in the 1st simplification step where the elements were broken down into bits (Fig. 4). As a result, alignments are possible where the bits from a single element of the 1st sequence are equivalenced to bits from several elements in the 2nd sequence and *vice versa* (Fig. 6).

For these 2 reasons, the function that is actually minimized is not the simple function P, but a related function E:

$$E = P + a \times \sum_{\substack{i=1 \\ V_i \neq 0}}^{N_a} \begin{pmatrix} \text{the number of crossings over the} \\ \text{2 equivalence lines of the relation } i \end{pmatrix}$$
$$+ b \times \left[\sum_{i=1}^{N_a^e} \begin{pmatrix} \text{number of seq. 2 elems. with bits} \\ \text{equivalent to bits of elem. } i \text{ of seq. } 1^{-1} \end{pmatrix} + \sum_{i=1}^{N_b^e} \begin{pmatrix} \text{number of seq. 1 elems. with bits} \\ \text{equivalent to bits of elem. } i \text{ of seq. } 2^{-1} \end{pmatrix} \right] . \tag{20}$$

The additional 2 terms can be viewed as constraints that force the solution to avoid the 2 undesired features described above. The parameters a and b are weights measuring the relative importance of these 2 constraints terms. Parameters N_a^e and N_b^e are the numbers of equivalenced elements (elem.) that contribute bits to the 1st and 2nd sequence (seq.), respectively.

The 4th and last slot of the simulated annealing technique to be filled in is the annealing schedule, which tells us how the actual minimization is carried out. It is described in the flowchart of Fig. 7.

In general, there may be a large number of different alignments each corresponding to one of the many local minima with approximately global minimum score. This indicates that the probabilistic representation of the alignment may be a more accurate description of the equivalences between the 2 sequences of binary relations. This can easily be obtained by repeating the simulated annealing minimization, each time with a different random seed, counting how many times the ith bit from the 1st sequence was equivalenced to the jth bit of the 2nd sequence and storing the results into the $(2 \times N_a)$ by $(2 \times N_b)$ bit frequency matrix $\mathscr{B}$.

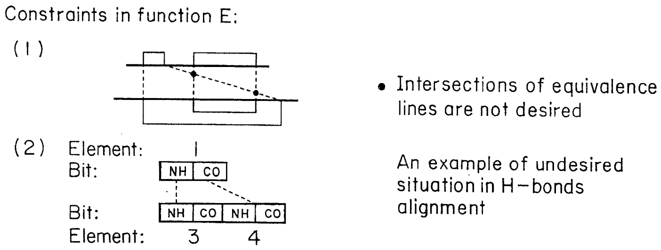

Constraints in function E:
(1)
• Intersections of equivalence lines are not desired
(2)
Element:
1
Bit:
NH
CO
Bit:
NH
CO
NH
CO
Element:
3
4
An example of undesired situation in H-bonds alignment

Figure 6. Definition of constraints terms in the energy function of the simulated annealing optimization. (1) The thick horizontal line represents the sequence of elements, the thin line stands for a relationship and the broken line indicates the equivalence. The 1st constraint then prevents violations of a "progression rule" which states that, for elements i and k from the 1st sequence and elements j and l from the 2nd sequence, if element i is equivalenced to element j, if element k is equivalenced to element l and if k is greater than i, then also l must be greater than j. (2) The 2nd constraint prevents situations such as that shown, where the amide and carbonyl group of the same residue in the 1st sequence are equivalenced to the amide and carbonyl groups that belong to 2 different residues in the 2nd sequence.

The pairs of bits with high B_{ij} values are more certain to be real equivalences than those pairs with lower values. If there is an uncertainty in some assignments it is indicated by a distribution of occurrences over the possibilities. Subsequently, the chances of random noise spoiling the alignment are smaller. It may also be noted that the inability to obtain the global optima in simulated annealing minimization is not critical, since the comparison of protein structures is founded on the parsimony principle and every solution with a score close to the global minimum should be helpful.

The last step of the simulated annealing method consists of converting the bit frequency matrix $\mathscr{B}$ into the element frequency matrix $\mathscr{U}^{\mathrm{r}}$:

$$U^{\mathrm{r}}_{e_i, e'_j} = \sum_{i=1}^{2N_a} \sum_{j=1}^{2N_b} B_{ij}. \tag{21}$$

Indices e_i and e'_j are element indices of the ith and jth bit from the 1st and 2nd bit sequence, respectively. The similarity matrix elements U^{r}_{ij} are then transformed into distance weights w^{r}_{ij} (eqn (18)) that are used directly in the dynamic programming procedure (eqns (2) and (3)).

It may be noted that this simulated annealing algorithm, being a generalization of a type of Needleman & Wunsch (1970) comparison, can align structures on the basis of a number of relations as well as properties at all levels of hierarchy simultaneously. Properties and hierarchy aspects could be introduced by modifying the $\mathscr{R}$ matrix according to the information in the $\mathscr{W}$ matrix, which would be computed without the relations terms (eqn (1)). However, we have not used this approach because the computer time required would be prohibitive when trying to accomplish simultaneous alignment of several structures (section (e), below), the explicit treatment of gap penalties (section (d)(i), below) and the numerous runs of the computer program to obtain align-

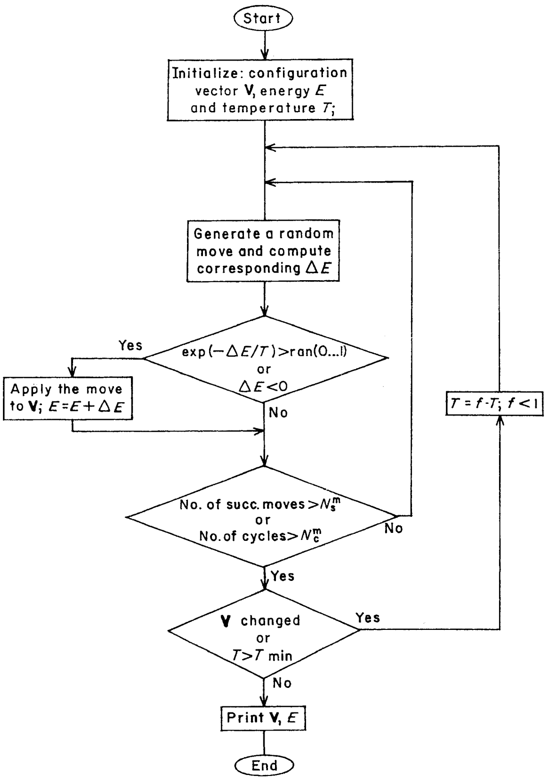

Start
Initialize: configuration vector V, energy E and temperature T;
Generate a random move and compute corresponding ΔE
exp(−ΔE/T)>ran(0...1) or ΔE<0
Yes
No
Apply the move to V; E=E+ΔE
No. of succ. moves > N_s^m or No. of cycles > N_c^m
No
Yes
V changed or T>T min
Yes
No
T = f·T; f<1
Print V, E
End

Figure 7. A block diagram of the simulated annealing minimization. In our implementation of simulated annealing optimization (Kirkpatrick *et al.*, 1983) the energy function is defined in eqn (20). A configuration vector and possible moves are defined in Fig. 5. One starts by initializing the configuration vector **V**, which represents the pairwise alignment of 2 sequences of relationships, by setting all elements of **V** to 0. Additionally, the energy corresponding to this alignment is calculated and the temperature T is also set to some initial value (usually 0·70). Then, the double loop of applying random changes to the alignment is entered. This double loop consists of the outer loop where, in every cycle, the temperature is decreased by a predetermined factor (usually 0·90) and of the inner loop where the random moves are generated and possibly applied to the configuration vector. In the inner loop, a corresponding energy change is calculated for every randomly generated move. This energy change is then subjected to a semi-random rest depending on the magnitude and sign of the change and on a current temperature (ran(0 . . .1) is a function returning a value for a random variable evenly distributed between 0 and 1). If the test is passed the move is applied to the alignment vector **V**. The inner loop exits either when it is executed more than N_c^m times (usually 150 times the number of elements in **V**) or when the number of accepted moves is more than N_s^m (usually 15 times the number of elements in **V**). The double loop is terminated when no changes to **V** are made in the last 2 temperature cycles or when temperature falls below a preset cutoff (usually 0·20). In short, the schedule of simulated annealing minimization consists of proposing a large number of random moves that may or may not be accepted. As the process continues, the unfavourable changes that increase the energy are less and less frequently accepted, as opposed to the changes that decrease the energy, which are always accepted. This process leads, one hopes, to the minimum of the energy function. Our implementation of the simulated annealing schedule follows suggestions by Press *et al.* (1986). succ., successful.

ments corresponding to different relative weighting of individual features.

In the application of the simulated annealing algorithm to the alignment of hydrogen bonds, a single amino acid residue is an element. The residues are subdivided into carbonyl and amide groups, which are the bits of the simulated annealing algorithm. The relation weight matrix $\mathscr{R}$ describing the differences between the 2 sequences of binary relations is constructed from the following Table of weights:

	CO . . NH	NH . . CO
CO . . NH	0	1
NH . . CO	1	0

where the 1st CO or NH group in the descriptor of the hydrogen bond type has a lower sequence number than the 2nd group. The weights a and b for the constraints in eqn (20) are 1·5 and 0, respectively.

When the hydrophobic contacts are aligned, the residues are elements that are subdivided into bits in such a way that every bit accounts for 1 hydrophobic contact. The weight matrix elements R_{ij} are all 0.

(d) *Structural comparisons*

(i) *Pairwise comparison*

The residue-by-residue weights W_{ij} from eqn (1) can be used directly in the sequence alignment algorithm of Needleman & Wunsch (1970) to obtain the comparison of 2 protein structures.

The problem of the optimal alignment of 2 sequences as addressed by the algorithm of Needleman & Wunsch is as follows. We are given 2 sequences of elements and an $M \times N$ weight matrix $\mathscr{W}$, where M and N are the numbers of elements in the 1st and 2nd sequence. The weight matrix is composed of weights W_{ij} describing differences between elements i and j from the 1st and 2nd sequence, respectively. The goal is to obtain an optimal set of equivalences that match elements of the 1st sequence to the elements of the 2nd sequence. The equivalence assignments are subject to the following "progression rule": for elements i and k from the 1st sequence and elements j and l from the 2nd sequence, if element i is equivalenced to element j, if element k is equivalenced to element l and if k is greater than i, then l must also be greater than j. The optimal set of equivalences is that with the smallest score. The score is a sum of weights corresponding to matched elements, also increased for occurrences of non-equivalenced elements (i.e. gaps). For a detailed discussion on this and related problems, see Sankoff & Kruskal (1983).

We will summarize the dynamic programming formulae used in our programs to obtain the optimal alignment, since they differ slightly from those already published (Sellers, 1979; Gotoh, 1982). The recursive dynamic programming formulae that give a matrix $\mathscr{D}$ are:

$$D_{i,j} = \min \begin{cases} P_{i,j} \\ D_{i-1,j-1} + W_{i,j} \\ Q_{i,j} \end{cases}$$

$$P_{i,j} = \min \begin{cases} D_{i-1,j} + g(1) \\ P_{i-1,j} + u \end{cases} \tag{22}$$

$$Q_{i,j} = \min \begin{cases} D_{i,j-1} + g(1) \\ Q_{i,j-1} + u, \end{cases}$$

where $g(l)$ is a linear gap penalty function:

$$g(l) = u \times l + v. \tag{23}$$

The uppermost formula in eqn (22) is calculated for $i = M$ and $j = N$. Variable l is a gap length and parameters u and v are gap penalty constants.

The arrays $\mathscr{D}$, $\mathscr{P}$ and $\mathscr{Q}$ are initialized as follows:

$$\begin{aligned} D_{i,0} &= \begin{cases} 0, & i \leq e \\ g(i), & e < i \leq N \end{cases} \\ D_{0,j} &= \begin{cases} 0, & j \leq e \\ g(j), & e < j \leq N \end{cases} \\ P_{i,0} &= Q_{i,0} = \infty, \quad i = 1, 2, \ldots, M \\ P_{0,j} &= Q_{0,j} = \infty, \quad j = 1, 2, \ldots, N, \end{aligned} \tag{24}$$

where parameter e is the maximal number of elements at sequence termini that are not penalized with a gap penalty if not equivalenced. A segment at the terminus of length e is termed an overhang. Note: there is a difference from the method of Gotoh (1982) in the initialization of the $\mathscr{P}$ and $\mathscr{Q}$ arrays.

The minimal score $d_{M,N}$ is obtained from:

$$d_{M,N} = \min(D_{i,N}, D_{M,j}), \tag{25}$$

where $i = M, M-1, \ldots, M-e$ and $j = N, N-1, \ldots, N-e$ to allow for the overhangs. The equivalence assignments are obtained by backtracking in matrix $\mathscr{D}$. Backtracking starts from the element $D_{i,j} = d_{M,N}$.

(e) *Multiple structural comparisons*

In the discussion given in the previous section, we have assumed that the 3-dimensional structures would be compared in a pairwise manner. However, such pairwise comparisons of several related proteins may not be self-consistent, i.e. the following transitivity rule can be broken. If residue a from protein A is equivalent to residue b in protein B, which in turn is equivalent to residue c in protein C, then the residue a from protein A must also be equivalent to residue c from protein C. This property is not always attained in the set of usual pairwise comparisons relating a group of similar proteins. For this reason we proceed by simultaneously aligning all structures.

Recently, several methods for efficient multiple sequence alignment have been described (Hogeweg & Hesper, 1984; Barton & Sternberg, 1987; Feng & Doolittle, 1987). The essence of these approaches is in the way of overcoming the computationally too exhaustive dynamic programming algorithm for finding the *global* minima of the multiple alignment. We adopt here a combination of these approaches.

The procedure is divided into 2 parts. In the 1st module, a dendrogram is constructed from pairwise comparisons of protein structures. Alternatively, a tree can be specified in a subjective way based on *a priori* knowledge about the relationships between the proteins of interest. The 2nd part then involves a stepwise incorporation of proteins, as imposed by the tree topology, into the growing multiple alignment.

(i) *Tree construction*

The comparison program described above is used to compute scores for all $N(N-1)/2$ pairwise comparisons of N proteins in a group. These scores are represented in a symmetrical $N \times N$ distance matrix that can be used directly as the input to a dendrogram-constructing program. We use the program KITSCH from the clustering package PHYLIP (Felsenstein, 1985). This program is based on the procedure by Fitch & Margoliash (1967), which constructs the tree by minimizing the $\sum_{i,j}[(d_{ij}-t_{ij})^2/d_{ij}^2]$, where d_{ij} is a distance for a pair (i, j) from an input distance matrix and t_{ij} is the corresponding distance obtained from a tree. Additionally, the sums of the branch lengths from the root of the tree to each of the leaves are constrained to the same value.

(ii) *Alignment*

The alignment procedure follows the tree structure, from the tips of the branches to the root. First, the 2 most similar proteins are aligned using the pairwise procedure described in section (d)(i), above, to give the 1st sub-alignment. Then either the 3rd protein is compared with the 1st sub-alignment or the 2nd most similar pair of proteins is compared in the usual pairwise way. Which of the 2 alternatives will be realized depends on the actual tree topology. Suppose the latter is the case and the tree then implies that the 2 already aligned pairs have to be joined next. This is again accomplished with the standard pairwise alignment technique (section (d)(i), above), except that the weight matrix $\mathscr{W}$ is defined by averaging the differences between the 2 groups of already aligned structures. In general, when 2 sub-alignments have to be compared, the weight matrix element W_{ij} is defined by:

$$W_{ij} = \frac{1}{n_1 \times n_2} \sum_{l=1}^{n_1} \sum_{k=1}^{n_2} \begin{cases} W^{lk}_{i'j'} & \text{if 2 residues are compared} \\ u^m \times u & \text{if a residue and a gap are compared} \\ 0 & \text{if 2 gaps are compared.} \end{cases} \tag{26}$$

Indices i and j are indices of the 2 positions compared, for the 1st and 2nd sub-alignment, respectively. Subscripts i' and j' are true indices for residues from proteins l and k, occupying sub-alignment positions i and j, respectively. $W^{lk}_{i'j'}$ is an element of a familiar weight matrix $\mathscr{W}$ defined in eqn (1) for a pairwise comparison of proteins l and k. Parameter u is the 1st gap penalty constant (eqn (23)) and u^m is a scaling factor (usually 2) that favours the alignment of gaps with gaps if greater than 1. Parameters n_1 and n_2 are the numbers of proteins in the 1st and 2nd sub-alignment, respectively. This definition of a weight matrix for a multiple comparison also allows the use of information from a pairwise alignment of relationships.

If the number of all proteins is N, $N-1$ alignments must be made to obtain the final multiple comparison. It may be noted that once an equivalence or gap is introduced it is not changed in later stages.

The tree-like addition of proteins (Hogeweg & Hesper, 1984; Feng & Doolittle, 1987) implemented here is contrasted to its special case, a simple sequential addition (Barton & Sternberg, 1987). The rationale behind this choice is that the most similar structures, and therefore the ones that are compared most accurately, are aligned first. In addition to a better alignment of similar structures, this also results in improved performance in later stages, since superior sub-alignments are used. Our experience is that the tree-like addition performs slightly better than the simple sequential addition, especially when one aligns a group of proteins with a wide range of similarities.

(f) *Classification of protein structures*

Following the approach of Johnson *et al.* (1989, 1990), we use the multiple structural alignment (section (e), above) to derive the classification of the aligned protein structures.

We begin with a calculation of the matrix of distance scores for all pairwise combinations of the proteins in a

Table 3
Structures of aspartic proteinases used for the comparison

Protein	Brookhaven code	No. of residues	Domain boundary	Resolution (Å)	R-factor (Å)	Reference
Endothiapepsin	4APE	178, 152	174–175	2·1	0·16	Blundell *et al.* (1985)
Rhizopuspepsin	2APR	178, 147	178–179	1·8	0·14	Suguna *et al.* (1987)
Penicillopepsin	2APP	174, 149	174–175	1·8	0·14	James & Sielecki (1983)
Porcine pepsin	†	174, 152	174–175	2·0	0·18	Cooper *et al.* (unpublished results)

† A code for pepsin structure used here is PEP.

group. We consider 2 types of pairwise distance scores. They are both obtained from the pairwise alignment implied by the multiple alignment. The 1st score, e', is obtained by summing the weights W_{ij} (eqn (1)) that relate the residues equivalent in the pairwise comparison. The 2nd type, a', includes gaps and is defined by the sum of e' and the gap penalties for the gaps (but not overhangs) in the pairwise alignment. Both scores, e' and a', are then normalized to give e and a; e' is divided by the number of equivalent residues in the pairwise comparison and a' is divided by the number of positions in the pairwise alignment (excluding overhangs). The final pairwise distance scores that are then used in the clustering procedure are:

$$\begin{aligned} E &= -100 \times \ln(1-e/D_c) \\ A &= -100 \times \ln(1-a/D_c), \end{aligned} \qquad (27)$$

where D_c is a comparison score for 2 unrelated proteins. We obtain this score by summing contributions from the features that were used to calculate the scores e and a. The contribution from any property is approximated by an average of the normalized residue-by-residue weights for the property (see eqn (2) for the definition of these weights). The 2nd type of contribution to the parameter D_c is associated with relationships and is obtained for every type of relationship from the simulated annealing alignment of 2 random sequences of relationships.

The matrix of pairwise distances is then used in the program KITSCH from the PHYLIP package (Felsenstein, 1985; section (e), above) to calculate the dendrogram that describes the classification of proteins considered.

Trees that classify different aspects of protein structure can be obtained by calculating pairwise scores E and A from different combinations of protein features. This is achieved by including in eqn (27) the element-by-element weights W_{ij} (eqn (1)) recalculated with the desired scaling factors ρ^f. Thus, features more variable in evolution, such as sequence identity, can be used for classification of similar proteins, and conserved features, such as hydrogen bonding, can be used for more divergent structures. Conversely, the clustering can also be used to infer the variability of a given protein feature in evolution.

(g) *Protein structures*

Protein structures from 3 protein families were used in this study to demonstrate the utility of the procedures described in the previous sections. These proteins include 10 serine proteinases of both mammalian and microbial origin, amino and carboxyl-terminal lobes of pepsin and 3 fungal aspartic proteinases and 6 mammalian, vertebrate, insect and plant globins. All structures, except for porcine pepsin (J. B. Cooper *et al.*, unpublished results), were taken from the October 1988 release of the

Table 4
Structures of serine proteinases and globins used for the comparisons

Proteins	Brookhaven code	No. of residues	Chain identifier	Resolution	R-factor (Å)	Reference
Serine proteinases:						
Bovine α-chymotrypsin	4CHA	239	A	1·68	0·23	Tsukada & Blow (1985)
Porcine elastase	3EST	240	—	1·65	0·17	Meyer *et al.* (1988)
Bovine trypsin (orthorhombic)	2PTN	223	—	1·55	0·19	Walter *et al.* (1982)
Rat tonin	1TON	227	—	1·80	0·20	Fujinaga & James (1987)
Rat mast cell protease	3RP2	224	A	1·90	0·19	Reynolds *et al.* (1985)
Porcine kallikrein	2PKA	232	A, B	2·05	0·22	Bode *et al.* (1983)
Streptomyces griseus trypsin	1SGT	223	—	1·70	0·16	Read & James (1988)
S. griseus proteinase A	2SGA	181	—	1·50	0·13	James *et al.* (1980)
S. griseus proteinase B	3SGB	185	E	1·80	0·13	Read *et al.* (1983)
L. enzymogenes α-lytic proteinase	2ALP	198	—	1·70	0·13	Fujinaga *et al.* (1985)
Globins:						
Human deoxy haemoglobin α-chain	2HHB	141	A	1·7	0·16	Fermi *et al.* (1984)
Human deoxy haemoglobin β-chain	2HHB	146	B	1·7	0·16	Fermi *et al.* (1984)
Sea lamprey haemoglobin V (cyano/met)	2LHB	149	—	2·0	0·14	Honzatko *et al.* (1985)
Sperm whale deoxy myoglobin	3MBN	153	—	2·0	0·23	Takano (1977)
Chironomous thummi thummi erythrocruorin	1ECD	136	—	1·4	0·19	Steigemann & Weber (1979)
Lupinus luteus leghaemoglobin	1LH1	153	—	2·0	—	Arutyunyan *et al.* (1980)

Brookhaven Protein Data Bank (Bernstein *et al.*, 1977). For protein names, codes, resolutions, crystallographic R-factors, original references and division of aspartic proteinases into individual lobes, see Tables 3 and 4. No ligands, domains and subunits were included in calculation of residue accessibilities for any of these families.

(h) *Computer programs*

Programs were written in Fortran 77 on a micro VAX II minicomputer running a VMS operating system and on a personal microcomputer running a UNIX system. A typical alignment of hydrogen bonding patterns of 2 aspartic proteinase lobes using the simulated annealing procedure takes roughly 30 min of micro VAX II central processor unit (c.p.u.) time and 5 min on a personal microcomputer (Intel 80386 processor, 80387 coprocessor running at 25 MHz and 8 MB RAM). Multiple comparison of 8 aspartic proteinase lobes using the dynamic programming algorithm (Fig. 11) takes about 12 min of micro VAX II c.p.u. time. The same comparison takes 4 min on the microcomputer. These statistics do not include the time required for calculation of solvent accessibilities, definition of secondary structures and multiple least-squares superposition of C^α atoms.

3. Results and Discussion

(a) *The comparison method*

The overall organization of the comparison program, COMPARER, is shown in Figure 8 for the first two levels in the hierarchy of the size of the protein building blocks: amino acid residues and secondary structure segments. At both levels, there are properties and relationships, including the spatial ones, between the elements of structure (Table 1). In fact, the level of structure is a measure of the simplification in the structural description. In larger, more complex proteins it may be useful to include an even higher level of structure by identifying motifs or domains; this will be particularly important if the program is used to make an automatic classification and clustering of all known protein structures.

The comparison of proteins using COMPARER involves several stages. In the first step, the elements of protein structure to be used in the comparison are defined. Additionally, various features associated with these elements must be recognized and the differences between these features defined. In the second stage, the segments such as α-helices and β-strands and features such as solvent accessibilities and hydrogen bonds are computed for all proteins to be aligned. In the third stage, the relationships are aligned in a pairwise manner using a simulated annealing procedure for all pairs of proteins. In the fourth stage, the weights obtained from the relationship alignments and from the differences in properties are merged to give the residue-by-residue weight matrix that is finally used in the familiar Needleman & Wunsch (1970) type of dynamic programming procedure to obtain a pairwise or a multiple alignment of proteins in question.

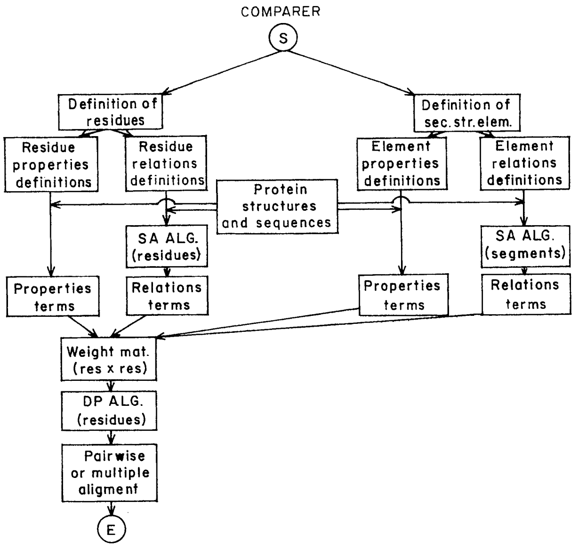

COMPARER
S
Definition of residues
Definition of sec. str. elem.
Residue properties definitions
Residue relations definitions
Element properties definitions
Element relations definitions
Protein structures and sequences
SA ALG. (residues)
SA ALG. (segments)
Properties terms
Relations terms
Properties terms
Relations terms
Weight mat. (res x res)
DP ALG. (residues)
Pairwise or multiple aligment
E

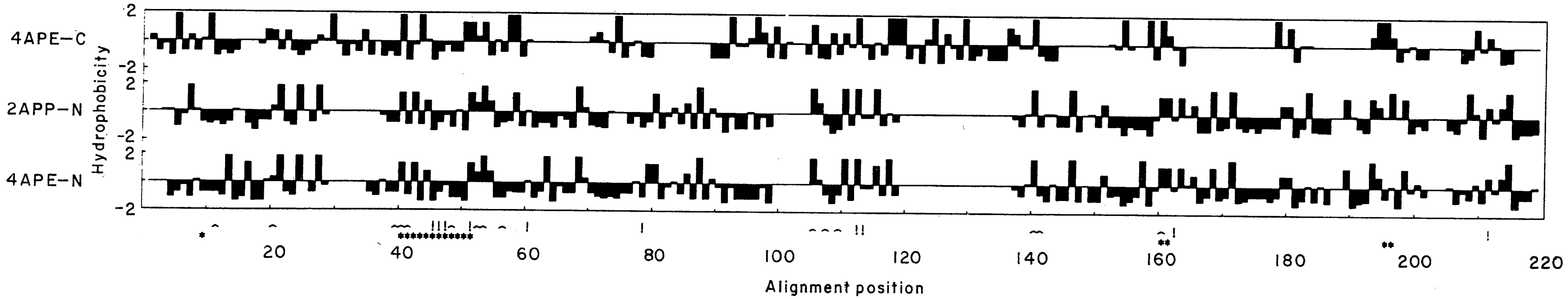
Figure 8. A block diagram of COMPARER. One starts with a definition of the elements (elem.) and the levels of protein structure that may be considered in the comparison. In this scheme, only residue and secondary structure (sec. str.) levels are indicated, although the diagram could easily be extended to the right to include higher levels, such as motifs, as well. The 2nd step is the definition of protein features, such as solvent accessibility, residue type and hydrogen bonding, which may be used in the comparison. These 2 steps are general definitions and do not concern specifically the proteins to be aligned. Next, the data on specific proteins are included and used to calculate the properties and relationships weights w_{ij}^{f}. These weights are then used in eqns (1) to (3) and possibly (26) to obtain the residue-by-residue weight matrix $\mathcal{W}$, either for a pairwise or a multiple comparison. Whereas the calculation of weights for properties is straightforward (Methods, sections (b)(i) to (b)(xiii)), the definition of weights for relationships requires the simulated annealing procedures (SA ALG.) (Methods, section (c)). Finally, the dynamic programming algorithm (DP ALG.) is applied on the weight matrix $\mathcal{W}$ to derive the pairwise or multiple alignment of proteins on the basis of features selected for the derivation of the weight matrix $\mathcal{W}$.

The scheme in Figure 8 emphasizes the dualistic nature of COMPARER. The most straightforward application is for the comparison of proteins. However, COMPARER can also be used to test hypotheses about the conserved aspects of protein structure. More precisely, since the input to the program consists of a series of hypotheses about conserved aspects of protein structure (i.e. definitions of elements, properties and relationships at several levels of protein structure), an evaluation of the final alignment with respect to a reference alignment provides a test for variability and conservation of the protein aspects used in the comparison. This is important for knowledge-based protein modelling, since it is the manipulation of the conserved properties and relationships of protein structure that leads to the prediction of the structure for the sequence of the unknown.

To illustrate and clarify this second and more

```
                   10        20        30        40        50
                 1       10        20              30
4APE-N        STGSATTTPIDSLDDAYITPV-QIG------TPAQTLNLDFDTGSSD
2APP-N       AASGVATNTPTA-NDEEYITPV-TIG--------GTTLNLNFDTGSAD
4APE-C     YTGSITYTAVSTKQ---GFWEWTSTGYAVGSGTFK-STSIDGIADTGTTL
               180           190       200       207       220
                     ^        ^                  ^^^   !!!^
                   *                              ***********

                   60        70        80        90       100
            40        50        60         70        80
4APE-N     LWVFSSETTASEVDGQTIYTPSKSTTAKLLSGATWSISYGDGSSSSGD--
2APP-N     LWVFSTELPASQQSGHSVYNPSA--TGKELSGYTWSISYGDGSSASGN--
4APE-C     LYLPATVVSA---------YWAQVSGAKS---------SSSVGGYVFPCS
                   230                        240       250
           !^^  ^   !                 !
           *

                  110       120       130       140       150
                    90                        100        110
4APE-N     ----VYTDTVSVGGLTVT------------------GQAVESAKKVS-SS
2APP-N     ----VFTDSVTVGGVTAH------------------GQAVQAAQQIS-AQ
4APE-C     A--TLPSFTFGVGSARIVIPGDYIDFGPISTGSSSCFGGIQSS-------
                    260       270       280       289
               ^ ^ ^  !!                          ^^

                  160       170       180       190       200
                  120       130        140         150
4APE-N     FTEDSTIDGLLGLAFSTLNTVSPTQQKTFFDNAKAS--LDSPVFTADLGY
2APP-N     FQQDTNNDGLLGLAFSSINTVQPQSQTTFFDTVKSS--LAQPLFAVALKH
4APE-C     -AGIG--INIFGD--------------VALKAA---------FVVFNGAT
                  300                     310
                    ^ !
                    **                                **

                   210
               162     170
4APE-N     H---APGTYNFGFIDTTA
2APP-N     Q---QPGVYDFGFIDSSK
4APE-C     T-----PTLGFASK
                 320   326
                     !
```

(a)



(b)

Fig. 9.

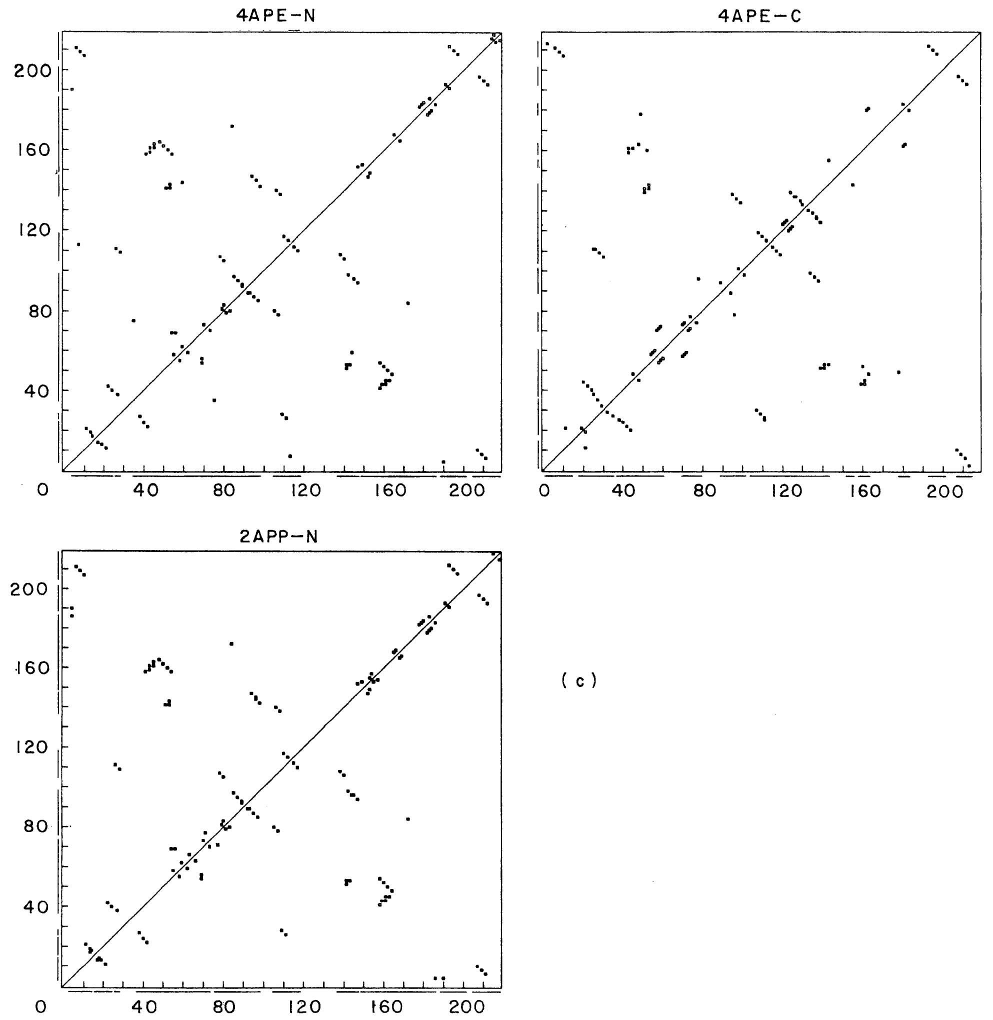
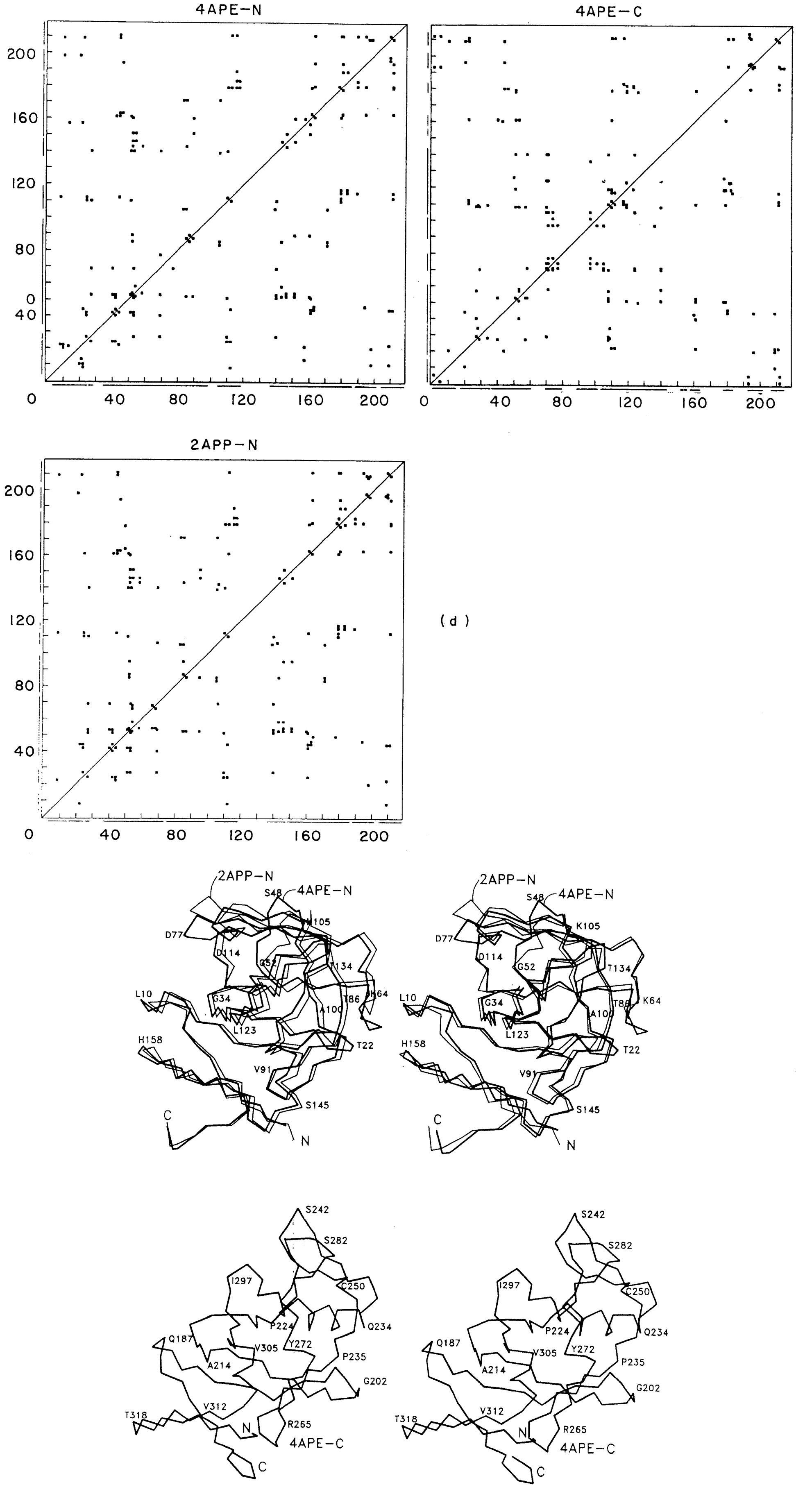


Fig. 9.

fundamental aspect of COMPARER, we show in Figure 9(a) to (f) six different representations of three protein structures that share the same fold: the amino-terminal lobes of endothiapepsin and penicillopepsin and the carboxyl-terminal lobe of endothiapepsin. While sequence identity between the amino-terminal lobes is 58%, the similarity between any of the amino-terminal lobes and carboxyl-terminal lobe is not statistically significant at the level of 3σ (Johnson *et al.*, 1990).

All representations in Figure 9(a) to (f) describe a protein, but each of these representations concentrates on a different aspect of protein structure. Thus, in Figure 9(a) the amino acid sequences of the three lobes are shown. This characterization of protein sequence is then modified by associating with every residue a hydrophobicity index and plotting this parameter (not the residue type as in a previous Figure) against the residue number (Fig. 9(b)). This representation shows more clearly the variation of hydrophobicity along the chain than does the primary structure (Fig. 9(a)). Note that in both characterizations the resemblance of amino-terminal lobes is evident, whereas the similarity between either of the amino-terminal lobes with a carboxyl-terminal lobe cannot be observed either by eye, or by rigorous statistical analysis of sequence alignments (Johnson *et al.*, 1990).

In Figure 9(c) the hydrogen bonding plots are shown: these plots were obtained by placing a marker at co-ordinates (i, j) of the symmetrical $N \times N$ distance plot if residues i and j are in a hydrogen bonding relationship (see Methods for a definition of a hydrogen bond); the filled square is used if an element i is an amide group and an empty square is used if an element i is a carbonyl group. The high conservation of hydrogen bonding patterns, as inferred from the comparison between all three lobes, suggests that the alignment of proteins that can incorporate information about the hydrogen bonding connectivities may also be successful for distantly related proteins. Distance plots of another type are shown in Figure 9(d); these are the hydrophobicity contact plots that were obtained by putting a marker at the element (i, j) of the distance plot if residues i and j are in the


4APE-N
4APE-C
2APP-N
200
160
120
80
0
40
0
40
80
120
160
200
2APP-N
S48
4APE-N
K105
D77
D114
G52
T134
L10
G34
T86
K64
A100
L123
H158
T22
V91
S145
C
N
S242
S282
I297
C250
P224
Q234
Q187
V305
Y272
P235
A214
G202
T318
V312
R265
4APE-C

(d)

(e)

Fig. 9.

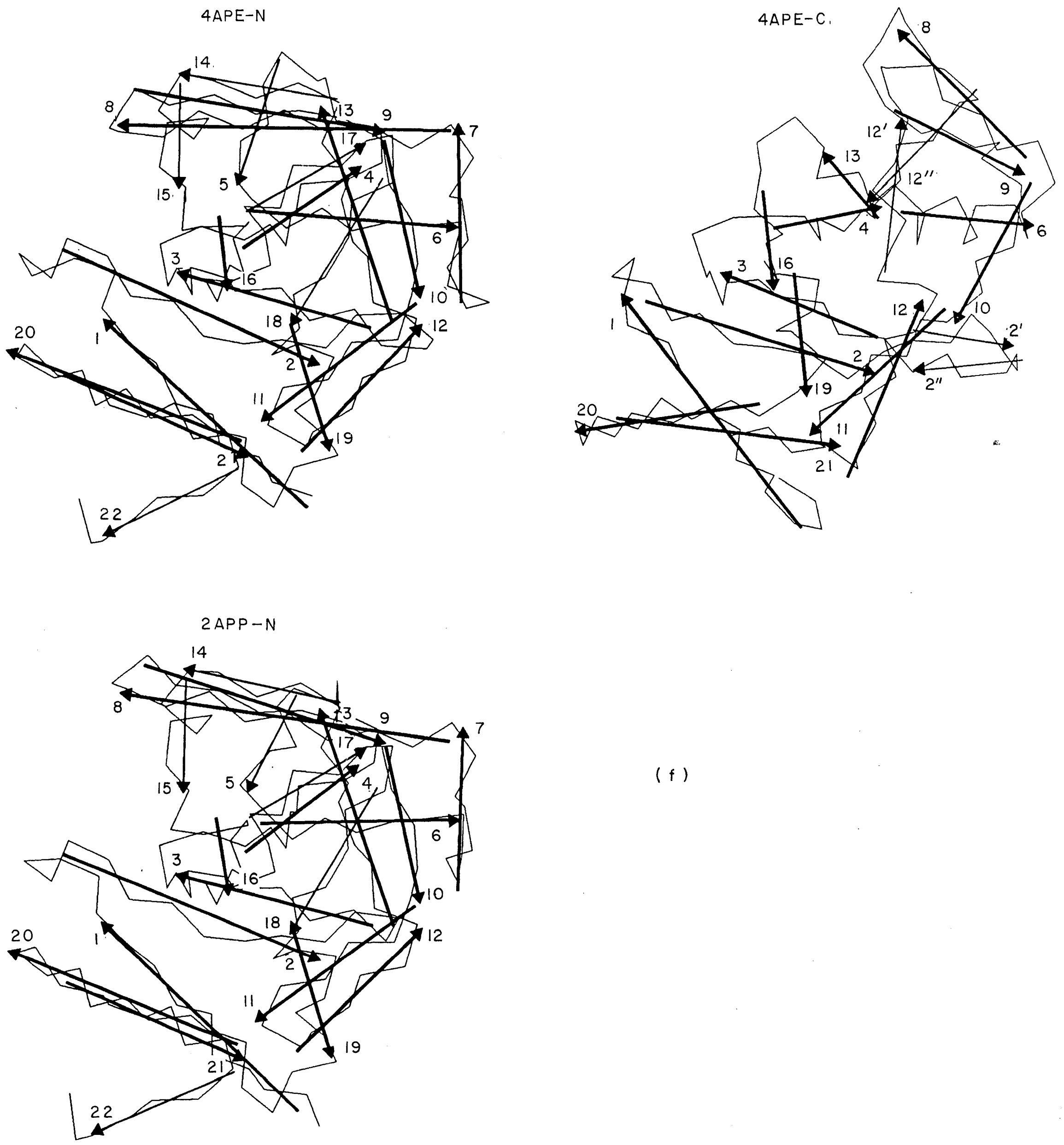
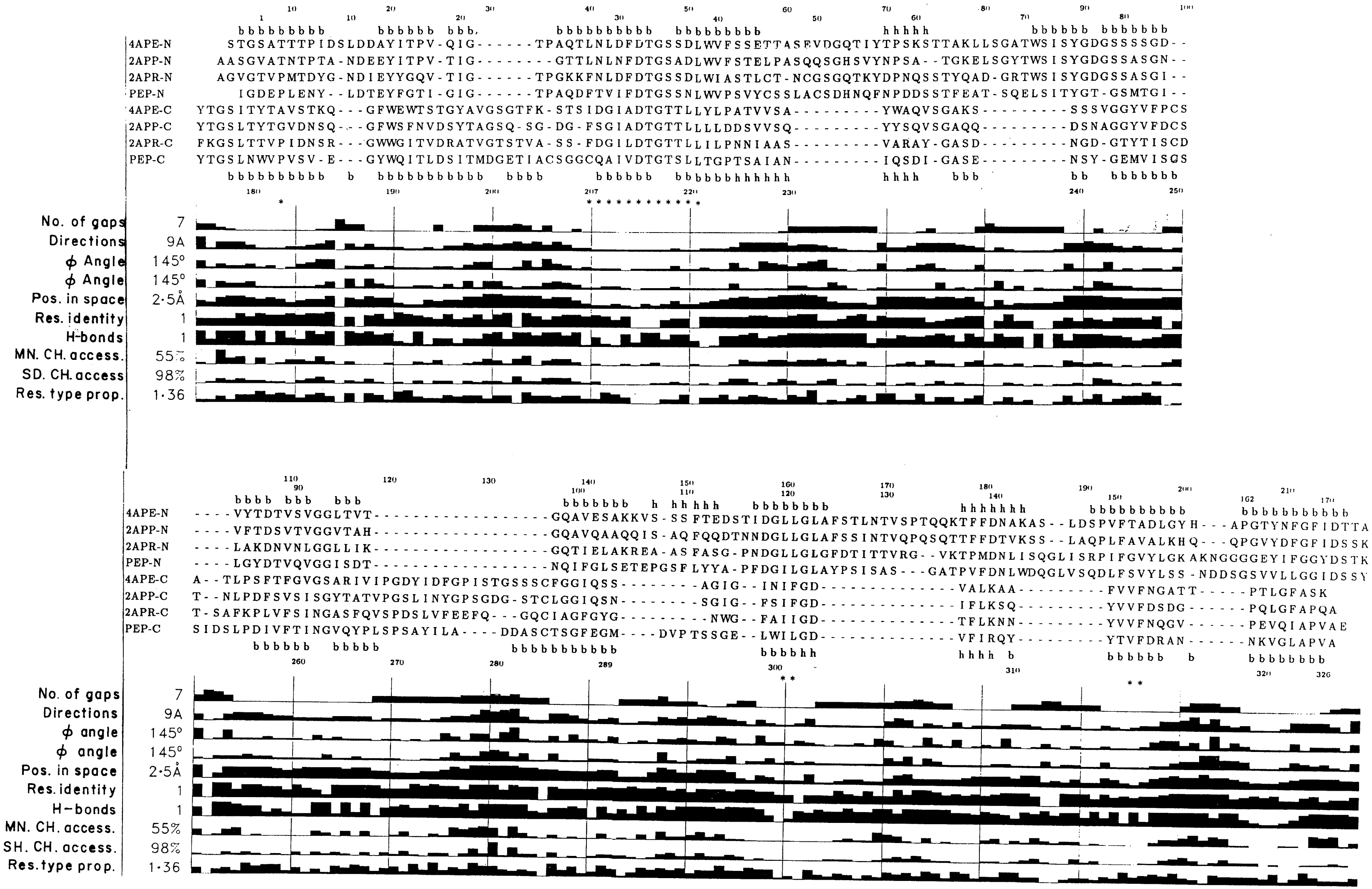


Figure 9. Different aspects of a protein that can be used simultaneously in the COMPARER to obtain the pairwise or multiple alignment. Three structures are selected as examples: amino-terminal lobes of endothiapepsin and penicillopepsin and carboxyl-terminal lobe of endothiapepsin. Each panel, (a) to (f), focuses on a different feature or aspect of the 3 lobes. The alignment from Fig. 10 was used to emphasize similarities between the lobes by introducing appropriate gaps. (a) Amino acid residue sequences. Three numbering schemes are adopted: the 1st line contains the alignment positions, the 2nd line contains the pepsin residue numbering for the amino-terminal lobe and the last line contains the pepsin numbering for the carboxyl-terminal lobe. Positions equivalenced by MNYFIT are indicated by stars, positions where all 3 residue types are identical are designated by an exclamation mark and those positions where all 3 residue types belong to one of the conserved groups (T, S), (V, L, I), (W, Y, F), (N, D) or (Q, E) are flagged by an arrow. For protein codes see Table 3. (b) Surrounding hydrophobicity plot. The hydrophobicity for each residue is plotted as a function of the alignment position. For designation of conserved positions see above. (c) Hydrogen bonding plot. A square is placed at the position (i, j) if residues i and j are hydrogen bonded. An empty square is used if residue i is a carbonyl group and residue j is an amide group; otherwise, a square is filled in. For the definition of a hydrogen bond see Methods, section (b)(iii). (d) Hydrophobic contacts plot. A square is placed at the position (i, j) if residues i and j are in a hydrophobic contact. For the definition of a hydrophobic relationship see Methods, section (b)(iii). (e) C^{α} backbones. (f) Helices, β-strands and extended segments superposed on the C^{α} backbones. The equivalent elements have the same indices. Thick lines are used for elements that occur in both the amino and carboxyl-terminal lobes. Segments with no equivalent counterparts are in thin lines. For the definition of the segments see Methods, section (b)(xiv).



```
                 10          20          30          40          50          60          70          80          90          100
              1          10          20          30          40          50          60          70          80
           bbbbbbbbb    bbbbbb bbb,            bbbbbbbbbb  bbbbbbbbb                   hhhhh            bbbbbb bbbbbbb
4APE-N     STGSATTTPIDSLDDAYITPV-QIG------TPAQTLNLDFDTGSSDLWVFSSETTASEVDGQTIYTPSKSTTAKLLSGATWSISYGDGSSSSGD--
2APP-N    AASGVATNTPTA-NDEEYITPV-TIG--------GTTLNLNFDTGSADLWVFSTELPASQQSGHSVYNPSA--TGKELSGYTWSISYGDGSSASGN--
2APR-N    AGVGTVPMTDYG-NDIEYYGQV-TIG------TPGKKFNLDFDTGSSDLWIASTLCT-NCGSGQTKYDPNQSSTYQAD-GRTWSISYGDGSSASGI--
PEP-N       IGDEPLENY--LDTEYFGTI-GIG------TPAQDFTVIFDTGSSNLWVPSVYCSSLACSDHNQFNPDDSSTFEAT-SQELSITYGT-GSMTGI--
4APE-C   YTGSITYTAVSTKQ---GFWEWTSTGYAVGSGTFK-STSIDGIADTGTTLLYLPATVVSA---------YWAQVSGAKS---------SSSVGGYVFPCS
2APP-C   YTGSLTYTGVDNSQ---GFWSFNVDSYTAGSQ-SG-DG-FSGIADTGTTLLLLDDSVVSQ---------YYSQVSGAQQ---------DSNAGGYVFDCS
2APR-C   FKGSLTTVPIDNSR---GWWGITVDRATVGTSTVA-SS-FDGILDTGTTLLILPNNIAAS---------VARAY-GASD---------NGD-GTYTISCD
PEP-C    YTGSLNWVPVSV-E---GYWQITLDSITMDGETIACSGGCQAIVDTGTSLLTGPTSAIAN---------IQSDI-GASE---------NSY-GEMVISCS
           bbbbbbbbbb   b   bbbbbbbbbbb  bbbb          bbbbbb  bbbbbbhhhhhh          hhhh   bbb          bb  bbbbbbb
              180            190           200          207          220         230                        240        250
               *                                        ***********

              110          120          130          140          150          160          170          180          190          200          210
               90                                     100          110          120          130          140          150                162       170
           bbbb bbb   bbb                          bbbbbbb   h  hhhhh   bbbbbbbb                   hhhhhhh          bbbbbbbbbb          bbbbbbbbbbb
4APE-N   ----VYTDTVSVGGLTVT-----------------GQAVESAKKVS-SSFTEDSTIDGLLGLAFSTLNTVSPTQQKTFFDNAKAS--LDSPVFTADLGYH---APGTYNFGFIDTTA
2APP-N   ----VFTDSVTVGGVTAH-----------------GQAVQAAQQIS-AQFQQDTNNDGLLGLAFSSINTVQPQSQTTFFDTVKSS--LAQPLFAVALKHQ---QPGVYDFGFIDSSK
2APR-N   ----LAKDNVNLGGLLIK-----------------GQTIELAKREA-ASFASG-PNDGLLGLGFDTITTVRG--VKTPMDNLISQGLISRPIFGVYLGKAKNGGGGEYIFGGYDSTK
PEP-N    ----LGYDTVQVGGISDT-----------------NQIFGLSETEPGSFLYYA-PFDGILGLAYPSISAS---GATPVFDNLWDQGLVSQDLFSVYLSS-NDDSGSVVLLGGIDSSY
4APE-C   A--TLPSFTFGVGSARIVIPGDYIDFGPISTGSSSCFGGIQSS--------AGIG--INIFGD---------------VALKAA----------FVVFNGATT-----PTLGFASK
2APP-C   T--NLPDFSVSISGYTATVPGSLINYGPSGDG-STCLGGIQSN--------SGIG--FSIFGD---------------IFLKSQ----------YVVFDSDG------PQLGFAPQA
2APR-C   T-SAFKPLVFSINGASFQVSPDSLVFEEFQ---GQCIAGFGYG---------NWG--FAIIGD---------------TFLKNN----------YVVFNQGV------PEVQIAPVAE
PEP-C    SIDSLPDIVFTINGVQYPLSPSAYILA----DDASCTSGFEGM----DVPTSSGE--LWILGD---------------VFIRQY----------YTVFDRAN------NKVGLAPVA
             bbbbbb  bbbbb                bbbbbbbbbbb         b       bbbbhh                hhhh b          bbbbbb   b        bbbbbbbb
                 260         270          280        289                  300                 310                                320       326
                                                                          **                                  **
```

Figure 10. The COMPARER multiple alignment of 8 aspartic proteinase lobes. Three numbering schemes are adopted: the 1st line contains the alignment positions, the 2nd line contains the pepsin residue numbering for the amino-terminal lobe and the last line contains the pepsin numbering for the carboxyl-terminal lobe. Stars below some of the alignment positions indicate residues that were found to be equivalent in multiple rigid-body superposition using the program MNYFIT and a cutoff distance 3·5 Å. h and b stand for the helical and β-sheet residues in endothiapepsin amino and carboxyl-terminal lobes (upper and lower line, respectively) as obtained from the corresponding Brookhaven Protein Data Bank file. Protein features used in this alignment were residue type properties ($\rho^{2} = 0{\cdot}10$), side-chain accessibility ($\rho^{7} = 0{\cdot}15$), main-chain accessibility ($\rho^{10} = 0{\cdot}05$), hydrogen bonds ($\rho^{14} = 0{\cdot}80$), sequence identity ($\rho^{15} = 0{\cdot}10$), position in space ($\rho^{17} = 0{\cdot}15$), ϕ dihedral angle ($\rho^{18} = 0{\cdot}05$), ψ dihedral angle ($\rho^{19} = 0{\cdot}10$) and main-chain directions ($\rho^{20} = 0{\cdot}20$). The gap penalty constants u and v were 0·83 and 0·80. Multiple alignment gap penalty weight u^{m} was 2·1. Parameters α, β and γ for main-chain directions (feature 20) were 7, 2 and 6, respectively. The guiding tree for the multiple alignment was ((((4APE-N, 2APP-N), 2APR-N), PEP-N), (((4APE-C, 2APP-C), 2APR-C), PEP-C)). For the protein codes see Table 3. The histograms show the variability of the features used to obtain the alignment. The variability of any feature f for any position in the alignment is defined as the average of residue-by-residue weights ${}^{n}w_{ij}^{f}$ for all pairwise residue comparisons at this position. The values to the left of the y-axis are the maxima in the corresponding variability histograms.

hydrophobic relationship as defined in Methods. The comparison of the plots for the two amino-terminal domains suggests that the precise nature of the hydrophobic contacts is not a particularly conserved feature in protein evolution. This becomes even more clear from a comparison of the plots for any of the amino-terminal lobes with the carboxyl-terminal lobe plot.

Next, Figure 9(e) shows the C^α backbones of the three lobes. While the similarity between amino-terminal lobes is apparent, the amino and carboxyl-terminal lobes are very different. The distortions, rotations and translocations of secondary structure elements occur while preserving the general relationships between them (i.e. a fold), indicating that a simple and straightforward rigid-body superposition will fail while the more flexible method incorporating only local conformations may still work. While all previous Figures concentrated on the residue level in the protein hierarchy, Figure 9(f) shows the constellation of secondary structure elements and amplifies the view that proteins are hierarchical entities, consisting of segments of secondary structure that in turn consist of residues. The arrangement of secondary structural elements emphasizes the relative translations and rotations that were evident from comparison of C^α backbones.

Figure 9(a) to (f) also shows a basic difference between the comparison of properties and relationships. Whenever the properties are compared, the relevant protein representation assumes the form of the sequence (for example, a sequence of residues. and a sequence of C^α positions). In contrast, whenever relationships are considered, the relevant representation of the protein must assume the form of a two-dimensional plot. With this representation, the problem of the alignment of relationships can be reformulated as the question of what are the minimum numbers of column and row pairs to be deleted from the two compared plots that lead to the exact match between the two plots. A relationship intrinsically contains more information than a property: the information that two residues are hydrogen bonded to each other is more useful for a comparison than the information that each of the two residues is involved in a hydrogen bond with an unspecified partner.

To test the *ad hoc* choice of protein features (Table 1) and the algorithms developed here for their use in obtaining the alignment of distantly related proteins, we have used three protein families: aspartic proteinase lobes (8 lobes), serine proteinases (10 proteins) and globins (6 proteins). We addressed the question of which of the suggested protein features are generally conserved in protein evolution and therefore suitable for protein comparison using our method. At the more technical level, we asked which values for parameters defined in Methods (for example, feature weights ρ^f, feature cutoff values d_c^f and gap penalties u and v) are the most appropriate for a comparison of proteins to be used in protein modelling. This was achieved by the trial-and-error process consisting of a semi-automatic variation of parameters and comparison of results with reference alignments. The detailed analysis of a number of additional alignments, as well as a more systematic optimization of the most influential parameters (feature weights ρ^f and gap penalties), will be published in a separate paper.

(b) *The alignments*

(i) *The aspartic proteinases*

The aspartic proteinases are a family of monomeric β-sheet proteins consisting of two lobes whose sizes are approximately 175 for the amino-terminal and 150 residues for the carboxyl-terminal lobe. The pairwise sequence identity among the whole enzymes, pepsin, endothiapepsin, penicillopepsin and rhizopuspepsin, is roughly 40%. The aspartic proteinases probably evolved from an ancestor corresponding to a single lobe by gene duplication and fusion events (Tang *et al.*, 1978; Blundell *et al.*, 1979). Although sequence similarity between the two lobes is not statistically significant (sequence identity is between 8 and 14%) 43 residues of the amino-terminal lobe have topological equivalents in the carboxyl-terminal lobe of the highly refined endothiapepsin when the two lobes are superposed by the program MNYFIT (Sutcliffe, 1987*a*). However, Figure 9(c), which shows the hydrogen bonding plots for the endothiapepsin amino and carboxyl-terminal lobes, demonstrates that more than 43 residues are topologically equivalent; it also indicates that the hydrogen bonding relationships are more robust indicators of topological equivalence than a rigid body superposition. This observation influenced the approach taken here for comparison of analogous structures.

The COMPARER alignment of eight aspartic proteinase lobes is shown in Figure 10. In the comparison of the lobes of the same type, gaps are associated only with loops and β-bulges. At this level of similarity, there are virtually no differences between the COMPARER alignment and careful manual alignment of six fungal structures and over 40 sequences of other fungal, mammalian and viral aspartic proteinases (J. P. Overington, unpublished results).

On the other hand, the alignment of the amino with the carboxyl-terminal lobes shows large changes between the two folds. These changes include substantial differences in the lengths of secondary structure elements as well as an insertion of a helix 109–113 (pepsin numbering) in the amino-terminal lobes. Comparison of the COMPARER alignment with the alignments by Overington (unpublished results), Blundell *et al.* (1979) and by Pearl & Taylor (1987) shows a major difference in only one region, which starts at residue 42 and ends at residue 87 (amino-terminal lobe numbering). In the amino-terminal lobe, this region includes the β-hairpin loop that covers the active site cleft and is not present in the carboxyl-terminal lobe. However, the dissimilarities in this part of the two folds are

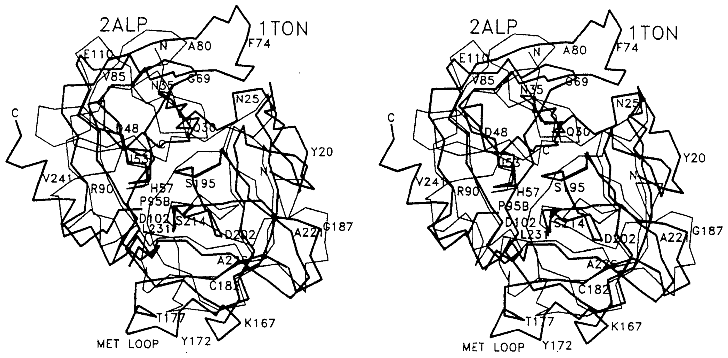


Figure 11. The stereo plot of the C^{α} backbones of tonin (thick line) and α-lytic protease (thin line). The superposition of these mammalian and microbial serine proteinases was obtained from the multiple least-squares fit of 10 serine proteinases by the program MNYFIT.

such that none of the four alignments can be clearly favoured over the other three. The second large difference between the COMPARER alignment and the manual alignment by Overington is a two-residue shift of a helix h'_n *versus* helix h'_c. In this case, the COMPARER alignment is recognized as a better choice, since the environments of the two helices in the COMPARER alignment are more similar.

(ii) *The serine proteinases*

The serine proteinases comprise a group of divergent β-sheet proteins for which six mammalian and four microbial three-dimensional structures have been defined by X-ray analysis (Fig. 11). They have been the subject of several comparative modelling studies (McLachlan & Shotton, 1971; Greer, 1981) and some mistakes have arisen as a result of the difficulty in the alignment of mammalian with microbial enzymes (for a discussion, see Delbaere *et al.*, 1975, 1979; James *et al.*, 1978; Read *et al.*, 1984). The sequence identities are between 26 and 55% for the pairwise comparisons within the group of trypsin-like serine proteinases that includes six mammalian enzymes and *Streptomyces griseus* trypsin. The percentage identities within the group of microbial serine proteinases (which includes the 3 remaining microbial enzymes) are 36 to 62%. However, sequence identity among the members from the two groups is considerably lower at 15 to 19%. We have experimented with superposition techniques (Sutcliffe *et al.*, 1987*a*) in modelling the serine proteinase domain of tissue-type plasminogen activator (Overington *et al.*, 1988) from the known serine proteinases. Only 57 positions out of approximately 200 are found to be topologically equivalent by multiple rigid-body superposition of all ten structures.

The COMPARER alignment of seven trypsin-like serine proteinases (Fig. 12) is very close to the proposed semi-manual structural alignments (James *et al.*, 1978; Read *et al.*, 1984; Overington *et al.*, 1988). The differences are limited to the shifts of few residues in the loop regions from one to the other side of the gaps.

The COMPARER alignment of the three microbial proteinases (Fig. 12) is also very similar to the structural alignment by James *et al.* (1978). However, in addition to the shifts of a residue or two at gap boundaries, a loop segment at position 115 (chymotrypsinogen A numbering) is aligned differently. Inspection of the three loops on a graphics terminal shows that the conformations and positions in space are so dissimilar that no convincing alignment can be obtained.

The comparison at the lowest level of similarity in the serine proteinase family, the alignment of the trypsin-like with the microbial proteinases, is a more challenging task. Nevertheless, in the COMPARER alignment (Fig. 12) all strands are aligned as suggested by Delbaere *et al.* (1979) on the basis of structures of α-lytic protease and elastase. However, apart from the shifts of boundary residues in the gap regions, other differences between the expert structural alignment (James *et al.*, 1978) and the COMPARER alignment do exist in the peripheral regions of the fold. The analysis of these differences gives some information about how the alignment is obtained by COMPARER and an expert. It must be noted, however, that some differences may be due to the improved quality of the crystallographic analyses available now compared to 1978.

The first four residues at the amino terminus of the microbial serine proteinases are aligned (James *et al.*, 1978) with the amino terminus of trypsin-like serine proteinases. Although there is a very convincing residue type matching, the local conformation, local environment and position in space for residues from position 18 onwards and the absence of the salt-bridge between the amino terminus and Asp194 in the microbial serine proteinases do not justify the large gap at this site.

The second interesting difference between the COMPARER comparison and that by James *et al.* (1978) is in the alignment of the methionine loop (residues 170 to 180). This region is a β-hairpin loop

```
               10        20        30        40        50        60        70        80        90        100
      1        10    20        30           40            50           60         70        80        90
                          bbbbbbbbbbb      bbbbbb bbbb     bbbb              bbbb            bbbbbbbbbbb
1ton             IVGGYKCEKNSQPWQVAVIN------EYLCGG-VLIDP---SWVITAAHCY---SN--NYQVLLGRNNLFKDEPFAQRRLVRQSFRHPD
2pka             IIGGRECEKNSHPWQVAIYH----YSSFQCGG-VLVNP---KWVLTAAHCK---ND--NYEVWLGRHNLFENENTAQFFGVTADFPHPG
2ptn             IVGGYTCGANTVPYQVSLNS-----GYHFCGG-SLINS---QWVVSAAHCY---KS--GIQVRLGEDNINVVEGNEQFISASKSIVHPS
4cha  CGVPAIQPVLSIVNGEEAVPGSWPWQVSLQDKT---GFHFCGG-SLINE---NWVVTAAHCG---VTT-SDVVVAGEFDQGSSSEKIQKLKIAKVFKNSK
3est             VVGGTEAQRNSWPSQISLQYRSGSSWAHTCGG-TLIRQ---NWVMTAAHCV---DRELTFRVVVGEHNLNQNNGTEQYVGVQKIVVHPY
3rp2             IIGGVESIPHSRPYMAHLDIVTEKGLRVICGG-FLISR---QFVLTAAHCK---GR--EITVILGAHDVRKRESTQQKIKVEKQIIHES
1sgt             VVGGTRAAQGEFPFMVRLS--------MGCGG-ALYAQ---DIVLTAAHCVSGSGNNTSITATGGVVDLQS--GAAVKVRSTKVLQAPG
2sga                        IAGGEAITTG-----GSRCSLGFNVSVNGVAHALTAGHCT---NISASWS------------------IGTRTGTS--
3sgb                        ISGGDAIYSS-----TGRCSLGFNVRSGSTYYFLTAGHCT---DGATTWWA----------NSARTTVLGTTSGSS--
2alp                      ANIVGGIEYSINN----ASLCSVGFSVTRGATKGFVTAGHCG---TVNATARI----------GG---AVVGTFAARV--
                            bbbbbbbbb      bbbbbbbbbbb   bbbbb  hhhh   hh bb'b                    bbbbbbb
                                  *           **                                                     *

               110       120       130       140       150       160       170       180       190       200
                  100       110            120       130       140         150       160         170
                     bbbbbb                              bbbbbbb                 bbbbbbbbhhhhhh  hhhhhhh
1ton  YI-PLPVHDH--SNDLMLLHLSEPA-DITGGVK----VIDLPT--KEPKVGSTCLASGWGSTNPS--EMVVSHDLQCVNIHLLSNEKCI--ETYKDNVT-
2pka  FNLSADGKDY--SHDLMLLRLQSPA-KITDAVK----VLELPT--QEPELGSTCEASGWGSIEPGPDDFEFPDEIQCVQLTLLQNTFCA--DAHPDKVT-
2ptn  YN----SNTL--NNDIMLIKLKSAA-SLNSRVA----SISLPT--SCASAGTQCLISGWGNTKSS--GTSYPDVLKCLKAPILSDSSCK--SAYPGQIT-
4cha  YN----SLTI--NNDITLLKLSTAA-SFSQTVS----AVCLPSASDDFAAGTTCVTTGWGLTRYA----NTPDRLQQASLPLLSNTNCK--KYWGTKIK-
3est  WN----TDDVAAGYDIALLRLAQSV-TLNSYVQ----LGVLPRAGTILANNSPCYITGWGLTRTNG---QLAQTLQQAYLPTVDYAICSSSSYWGSTVK-
3rp2  YN----SVPN--LHDIMLLKLEKKV-ELTPAVN----VVPLPSPSDFIHPGAMCWAAGWGKTGVR---DPTSYTLREVELRIMDEKACV--DYR--YYEY
1sgt  YN-----GT---GKDWALIKLAQPI-N----QP----TLKIAT--TTAYNQGTFTVAGWGANREG---GSQQRYLLKANVPFVSDAACR--SAYGNELVA
2sga  FP----------NNDYGIIRHSNPA-AADGRVYLYNGSYQDITTAGNAFVGQAVQRSGSTT--------------GLRSGSVTGLNATV---NYGSSGIV
3sgb  FP----------NNDYGIVRYTNTTIPKDGTVG-----GQDITSAANATVGMAVTRRGSTT--------------GTHSGSVTALNATV---NYGGGDVV
2alp  FP----------GNDRAWVSLT-SAQTLLPRVANG-SSFVTVRGSTEAAVGAAVCRSGRTT--------------GYQCGTITAKNVTA---NY-AEGAV
                       bbbbb      bbbbbb   bbbbb bbb     bbbbbbb                      bbbbbbbbb   bb   bbb
                   ********                               ******                  *******

               210       220       230       240       250       260       270       280
        180          190       200        210          220            230       240
       bbbbbb               bbbbbbb   bbbbbbbbbb                  bbbbbhhhhhhhhhhhhhhh
1ton  DVMLCAGEM-EGGKDTCAGDSGGPLICD-----GVLQGITSGGAT---PCAK----PKTPAIYAKLIKFTSWIKKVMKENP
2pka  ESMLCAGYL-PGGKDTCMGDSGGPLICN-----GMWQGITSWGHT---PCGS----ANKPSIYTKLIFYLDWIDDTITENP
2ptn  SNMFCAGYL-EGGKDSCQGDSGGPVVCS-----GKLQGIVSWGSG----CAQ----KNKPGVYTKVCNYVSWIKQTIASN
4cha  DAMICAG-A-SG-VSSCMGDSGGPLVCKKN-GAWTLVGIVSWGSS---TCST----S-TPGVYARVTALVNWVQQTLAAN
3est  NSMVCAG-G-DGVRSGCQGDSGGPLHCLVN-GQYAVHGVTSFVSR--LGCNV----TRKPTVFTRVSAYISWINNVIASN
3rp2  KFQVCVGSP-TTLRAAFMGDSGGPLLCA-----GVAHGIVSYGHP---DAK-------PPAIFTRVSTYVPWINAVIN
1sgt  NEEICAGYPDTGGVDTCQGDSGGPMFRKDNADEWIQVGIVSWGYG----CAR----PGYPGVYTEVSTFASAIASAARTL
2sga  YGMIQTN------VCAQPGDSGGSLFA-----GSTALGLTSGGS-----GNCR---TGGTTFYQPVTEALSAYGATVL
3sgb  YGMIRTN------VCAEPGDSGGPLYS-----GTRAIGLTSGGS-----GNCS---SGGTTFFQPVTEALVAYGVSVY
2alp  RGLTQGN------ACMGRGDSGGSWIT----SAGQAQGVMSGGNVQSNGNNCGIPASQRSSLFERLQPILSQYGLSLVTG
      bbbbbb                bbbbb      bbbbbbbbbbb     bb      bbbbbbbhhhhhhh
         ****        ***********          *****                   ****
```

Figure 12. The COMPARER multiple alignment of 10 serine proteinases. The 1st line contains the alignment positions, the 2nd line contains the numbering for chymotrypsinogen A, stars indicate residues found equivalent by the multiple superposition using the program MNYFIT. *h* and *b* stand for the helical and β-sheet residues in α-chymotrypsin (upper line) and *Streptomyces griseus* proteinase A (lower line) as obtained from the corresponding Brookhaven Protein Data Bank files. Protein features and their scaling factors ρ^f used in this comparison were residue type properties ($\rho^2 = 0{\cdot}20$), side-chain accessibility ($\rho^7 = 0{\cdot}10$), main-chain accessibility ($\rho^{10} = 0{\cdot}10$), hydrogen bonds ($\rho^{14} = 0{\cdot}60$), sequence identity ($\rho^{15} = 0{\cdot}20$), position in space ($\rho^{17} = 0{\cdot}30$), ϕ dihedral angle ($\rho^{18} = 0{\cdot}05$), ψ dihedral angle ($\rho^{19} = 0{\cdot}20$), and main-chain directions ($\rho^{20} = 0{\cdot}20$). Parameters α, β and γ for main-chain directions (feature 20) were 7, 2 and 6, respectively. The gap penalty constants u and v were 1·7 and 1·0. Multiple alignment gap penalty weight u^m was 2·2. The guiding tree for the multiple alignment was ((((((1ton, 2pka), 2ptn), (4cha, 3est)), 3rp2), 1sgt), ((2sga, 3sgb), 2alp)). For the protein codes see Table 4.

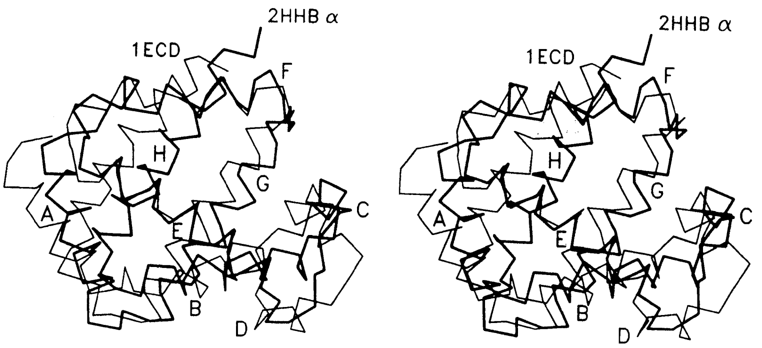


Figure 13. The stereo plot of the C^{α} backbones of erithrocruorin (thin line) and the α-chain of human haemoglobin (thick line). The superposition of these 2 insect and mammalian globins was obtained from the multiple least-squares fit of 6 globins by the program MNYFIT. The 8 helices are labelled with upper-case characters.

in the microbial serine proteinases, but includes a short helix in the trypsin-like serine proteinases. It also occupies a different volume when the structures are superposed. These two large differences result in the inability to favour any of the two alignments and imply that the alignment in this region is somewhat subjective.

The third significant difference includes the disulphide bridge between cysteine residues 191 and 220 in the trypsin-like serine proteinases. The alignment by James *et al.* (1978) makes this disulphide bridge equivalent to the disulphide bond C191–C220 in the microbial serine proteinases. However, the rigid-body superposition, local conformation and hydrogen bonding pattern lead COMPARER to make the first cysteine (C191) from the trypsin-like serine proteinases equivalent to the residue that is two positions after the cysteine C191 in the microbial serine proteinases; this is also recognized in the later alignment of β-sheets by James's group (Delbaere *et al.*, 1979). Having lost the structural equivalence of the first cysteine residues in the two disulphide bridges, the second cysteine residues do not have to be aligned either, which is what happens in the COMPARER alignment.

The last difference between the two alignments is in the carboxyl-terminal region. This region is α-helical in both groups of serine proteinases, except that in the microbial serine proteinases the helix is interrupted some five residues before the end of the chain and makes a right angle turn towards the bulk of the molecule. James *et al.* (1978) introduced a gap in the trypsin-like serine proteinases that is within the helices of both the microbial and trypsin-like serine proteinases, whereas the COMPARER alignment does not have any gap in this region.

(iii) *The globins*

While the pairwise sequence identities of structurally defined globins can be as low as 15%, all known structures conform to the same fold consisting of eight α-helices labelled A, B, C, D, E, F, G and H, with the exception of some structures that are missing the small D helix (Fig. 13). Gaps and changes in local conformation are always confined to the regions on the periphery of the molecule, helix termini and surface loops connecting the helices. Nevertheless, spatial relationships between the helices have changed considerably, resulting in the differences of as much as 7 Å and 30° in the relative position and orientation of helices when homologous pairs of helices are superposed (Lesk & Chothia, 1980). Consequently, the automated multiple rigid-body superposition of whole structures using the program MNYFIT (Sutcliffe *et al.*, 1987*a*) finds only 15 equivalent C^{α} atoms out of roughly 140 residues (Fig. 14). On the other hand, careful manual alignment identifies 116 structurally equivalent positions (Lesk & Chothia, 1980).

We compared six structures that include globins of mammalian, vertebrate, insect and plant origin to see whether the COMPARER is able to overcome the problems inherent to the rigid-body superposition. Since the main-chain hydrogen bonding relationships are of little use for the alignment of α-helical proteins, we were also interested in the quality of the alignment obtained only from the properties.

The COMPARER alignment is shown in Figure 14. The COMPARER alignment is virtually identical with the alignment by Bashford *et al.* (1987); the differences are limited to loop regions where shifts of one or two residues from one side of a gap to another occur.

(iv) *The summary of the alignments*

On the whole, it can be concluded from the alignments of three protein families that COMPARER is suitable for the automated comparison of rather divergent protein structures and that it can mimic human criteria for deriving the most parsimonious alignment. The key role is played by the hydrogen bonding information that, together with the equivalences from the rigid-body superposition, provides a greater number of anchor points from which the alignment is extended on the basis of other features. Thus, the equivalences between what one intuitively sees as framework regions are easily identified, even if large rigid-body shifts have occurred. The fraction of the whole fold that may be aligned with some confidence is probably larger than in any other comparison method published so far.

```
                10        20        30        40        50        60        70        80        90       100
                          10        20        30        40        50             60        70             80
2HHBα            VLSPADKTNVKAAWGKVGAHAGEYGAEALERMFLSFPTTKTYFPHF-DLSH-----GSAQVKGHGKKVADALTNAVAHVD--D---MPNAL
2HHBβ           VHLTPEEKSAVTALWGKV--NVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDAVMGNPKVKAHGKKVLGAFSDGLAHLD--N---LKGTF
3MBN             VLSEGEWQLVLHVWAKVEADVAGHGQDILIRLFKSHPETLEKFDRFKHLKTEAEMKASEDLKKHGVTVLTALGAILKKKG--H---HEAEL
1ECD              LSADQISTVQASFDKVKGD----PVGILYAVFKADPSIMAKFTQFAG-KDLESIKGTAPFETHANRIVGFFSKIIGELP--N---IEADV
2LHB    PIVDTGSVAPLSAAEKTKIRSAWAPVYSTYETSGVDILVKFFTSTPAAQEFFPKFKGLTTADELKKSADVRWHAERIINAVDDAVASMD--DTEKMSMKL
1LH1            GALTESQAALVKSSWEEFNANIPKHTHRFFILVLEIAPAAKDLFSFLKGTSEVP--QNNPELQAHAGKVFKLVYEAAIQLEVTGVVVTDATL
                                                                           ***************
                   AAAAAAAAAAAAA    BBBBBBBBBBBBBBBBCCCCC CC                 EEEEEEEEEEEEEEEEEEE   f   fffff

              110       120       130       140       150       160
               90       100       110       120       130       140
2HHBα   SALSDLHAHKLRVDPVNFKLLSHCLLVTLAAHLPAEFTPAVHASLDKFLASVSTVLTSKYR
2HHBβ   ATLSELHCDKLHVDPENFRLLGNVLVCVLAHHFGKEFTPPVQAAYQKVVAGVANALAHKYH
3MBN    KPLAQSHATKHKIPIKYLEFISEAIIHVLHSRHPGDFGADAQGAMNKALELFRKDIAAKYKELGYQG
1ECD    NTFVASHKPR-GVTHDQLNNFRAGFVSYMKAHT--DF-AGAEAAWGATLDTFFGMIFSKM
2LHB    RNLSGKHAKSFQVDPEYFKVLAAVIADTVAAG---------DAGFEKLMSMICILLRSAY
1LH1    KNLGSVHVSK-GVADAHFPVVKEAILKTIKEVVGAKWSEELNSAWTIAYDELAIVIKKEMDDAA

        ffFFFFFFFF  GGGGGGGGGGGGGGGGGGG    HHHHHHHHHHHH HHHHHHHH
```

Figure 14. The COMPARER multiple alignment of 6 globins. The 1st line contains the alignment positions, the 2nd line contains the numbering for the human haemoglobin α-chain, stars stand for the positions found equivalent by the multiple superposition using the program MNYFIT and the last line contains the residue assignments to individual helices for the α-chain of human haemoglobin (Lesk & Chothia, 1980). Protein features and their scaling factors ρ^f used in this comparison were local fold ($\rho^1 = 0{\cdot}10$), residue type properties ($\rho^2 = 0{\cdot}10$), main-chain orientation relative to molecular centre of gravity ($\rho^6 = 0{\cdot}10$), main-chain accessibility ($\rho^{10} = 0{\cdot}30$), sequence identity ($\rho^{15} = 0{\cdot}40$), ψ dihedral angle ($\rho^{19} = 0{\cdot}10$) and main-chain directions in space ($\rho^{20} = 0{\cdot}20$). Parameters α, β and γ for local fold (feature 1) and main-chain directions (feature 20) were 7, 2 and 6, respectively. Both gap penalty constants u and v were 1·0. Multiple alignment gap-penalty weight u^m was 1·0. The guiding tree for the multiple alignment was (((((2HHBA, 2HHBB), 3MBN), 1ECD), 2LHB), 1LH1). For the protein codes see Table 4.

However, the quality of the COMPARER alignment in divergent loop regions still cannot be properly assessed, primarily because no reliable reference alignments exist. It is possible that loop regions will have to be compared after an alignment of framework regions is fixed. Additionally, different features, weighting schemes and guiding trees may have to be used for the variable regions to obtain the desired alignments.

(c) *Clustering of protein structures*

The first systematic clustering of protein structures was described by Eventoff & Rossmann (1975). They constructed dendrograms based on structural features alone to describe distant phylogenetic relationships among the mononucleotide and dinucleotide binding proteins. In this paper, however, we are concerned principally with the clustering of proteins to assist in selection of appropriate structures for modelling.

Recently, Johnson *et al.* (1989, 1990) have shown that a useful structural pairwise distance metric can be defined from fractional topological equivalence and root-mean-square deviation, as calculated by least-squares superposition. They also show that this distance measure correlates well with the sequence metric. Our procedure extends this approach by reflecting in the classification additional structural and sequence features. Moreover, since these features can include relationships such as hydrogen bonding patterns, which are known to be conserved in evolution, structures that bear little similarity in other respects can be compared at statistically significant levels.

Figure 15 shows the classification of six globin structures based on the same features that were used to obtain the multiple alignment in Figure 14. This tree is identical in topology with the tree obtained on the basis of primary structure and similar to the tree from the rigid-body superposition (Johnson *et al.*, 1989, 1990). Such structural trees can be used in conjunction with the sequence-based trees to select automatically appropriate structures and fragments for modelling the sequence of the unknown fold (M. S. Johnson & T. L. Blundell, unpublished results).

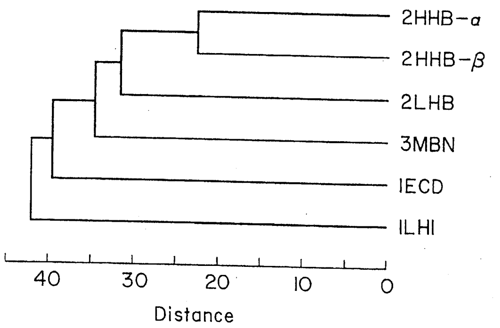


Figure 15. The classification of globin structures. The tree was constructed by the program KITSCH from the PHYLIP package as described in Methods, section (e). The same combination of features that was applied to calculate the multiple alignment in Fig. 14 was also used to derive this tree. The standard deviation for this tree is ±5%.

(d) *Generalized topological equivalence*

A consequence of using COMPARER for protein comparison is that the definition of the topological equivalence does not seem to be as straightforward as in the rigid-body superposition. For the superposition, topologically equivalent residues are defined as residues in the superposed structures that are within a certain distance of each other and obey the "progression rule". The result of the rigidity of this definition is a rapid decrease in the fraction of topologically equivalent residues when comparing more distantly related proteins (Johnson *et al.*, 1989). However, this is usually not due to a disappearance of all similarities between the regions that are not equivalenced. For example, the strands *a*, *b*, *q* and *r* of amino and carboxyl-terminal lobes of the aspartic proteinases (Fig. 9(c), (e) and (f), segment labels 1, 2, 21 and 22, respectively) retain equivalent hydrogen bonds, display the same general disposition and have a similar role in the whole structures, although the majority of C^{α} positions are too different to be identified as topologically equivalent by the MNYFIT program.

The reduction in the fractional number of topologically equivalent residues is especially disadvantageous for knowledge-based protein modelling where procedures for building equivalent regions are easier to devise (Sutcliffe *et al.*, 1987*a*) than those for modelling the non-equivalenced regions. Therefore, a generalized, more flexible definition of topological equivalence is required. Clearly, such a definition should be based on several different types of information and should not be limited only to the positional co-ordinates of the C^{α} atoms. It is also apparent that topological equivalence is not a discrete variable with only "yes" and "no" values; it is a variable with a continuum of values ranging from the most similar to the most different for every position in the alignment of two or more structures. Thus, we define the degree of the topological equivalence in any feature *f* for any position in the alignment as the mean of the normalized residue by residue weights ${}^{n}w^{f}_{ij}$ for all pairwise residue comparisons at this position (Fig. 10). Variability histograms are obtained by plotting the degree of topological equivalence as a function of the alignment position. In this paper, the same features on which the alignment is based are used to derive these variability histograms (Fig. 10).

Variability histograms are a simple representation of the variability and conservation of features in the given family fold. As such, they should prove to be useful in the analysis of invariants of protein structure. They are also helpful in assessing the quality of the alignment in a particular region of a family fold, since the quality is proportional to the local similarity of compared structures.

(e) *Conclusions*

A protein alignment method was developed that allows for systematic inclusion of a number of different types of information into the comparison of proteins. Most notably, relationships, as opposed to properties, are used for the first time in the automated derivation of topological equivalence. In addition, proteins can be treated as hierarchical entities; thus, information from different levels of protein structure can be considered in the alignment.

The comparison method has proved to be successful in the alignments of relatively dissimilar proteins from three families: globins of mammalian, vertebrate, insect and plant origin, aspartic proteinase lobes and mammalian and microbial serine proteinases.

The alignment approach described here facilitates an extensive structural comparison of all related proteins and motifs in the Brookhaven Protein Data Bank. From these alignments, some rules of protein structure can be inferred and used both for modelling by homology and for improving the comparison method itself.

We are indebted to our colleagues Dr Mark Johnson, Mr John Overington, Mr David Smith, Dr Mike Sutcliffe and Dr Geoff Barton for very useful discussions. We are also very grateful to Mr Jon Cooper for making available the structure of porcine pepsin prior to publication and to Mr John Overington for the manual alignment of aspartic proteinases and for his program for displaying multiple alignments. Additionally, we are thankful to an unknown referee for spotting several mistakes in the original manuscript. A.Š. is supported by an ORS Awards Scheme, the Research Council of Slovenia and the J. Stefan Institute, Slovenia.

References

Argos, P. (1987). *J. Mol. Biol.* **193**, 385–396.

Argos, P. & Rossmann, M. G. (1979). *Biochemistry*, **18**, 4951–4960.

Arutyunyan, E. G., Kuranova, I. P., Vainshtein, B. K. & Steigemann, W. (1980). *Kristallografiya*, **25**, 80–110 [in Russian].

Barton, G. J. & Sternberg, M. J. E. (1987). *J. Mol. Biol.* **198**, 327–337.

Bashford, D., Chothia, C. & Lesk, A. M. (1987). *J. Mol. Biol.* **196**, 199–216.

Bernstein, F. C., Koetzle, T. F., Williams, G. J. B., Meyer, E. F., Brice, M. D., Rodgers, J. R., Kennard, O., Shimanovichi, T. & Tasumi, M. (1977). *J. Mol. Biol.* **112**, 535–542.

Blundell, T. L., Sewell, B. T. & McLachlan, A. D. (1979). *Biochim. Biophys. Acta*, **580**, 24–31.

Blundell, T. L., Jenkins, J., Pearl, L. & Sewell, T. (1985). In *Aspartic Proteinases and Their Inhibitors* (Kostka, V., ed.), pp. 151–161, Walter de Gruyter, Berlin.

Blundell, T. L., Sibanda, B. L., Sternberg, M. J. E. & Thornton, J. M. (1987). *Nature (London)*, **326**, 347–352.

Blundell, T. L., Carney, D., Gardner, S., Hayes, F., Howlin, B., Hubbard, T., Overington, J., Singh, D. A., Sibanda, B. L. & Sutcliffe, M. (1988). *Eur. J. Biochem.* **172**, 513–520.

Bode, W., Chen, Z., Bartels, K., Kutzbach, C., Schmidt, G., Kastner, H. & Bartunik, H. (1983). *J. Mol. Biol.* **164**, 237–282.

Braun, W. & Go, N. (1985). *J. Mol. Biol.* **186**, 611–626.

Browne, W. J., North, A. C. T., Phillips, D. C., Brew, K., Vanaman, T. C. & Hill, R. L. (1969). *J. Mol. Biol.* **42**, 65–86.

Chothia, C. (1984). *Annu. Rev. Biochem.* **53**, 537–572.

Chothia, C. & Lesk, A. M. (1986). *EMBO J.* **5**, 823–826.

Chothia, C., Lesk, A. M., Levitt, M., Amit, A. G., Mariuzza, R. A., Phillips, S. E. V. & Poljak, R. J. (1986). *Science*, **233**, 755–758.

Crippen, G. M. (1977). *J. Comp. Phys.* **24**, 96–107.

Dayhoff, M. O., Barker, W. C. & Hunt, L. T. (1983). *Methods Enzymol.* **91**, 524–545.

Delbaere, L. T. J., Hutcheon, W. L. B., James, M. N. G. & Thiessen, W. E. (1975). *Nature (London)*, **275**, 758–763.

Delbaere, L. T. J., Brayer, G. D. & James, M. N. G. (1979). *Nature (London)*, **279**, 165–168.

Eventoff, W. & Rossmann, M. G. (1975). *CRC Crit. Rev. Biochem.* **3**, 111–140.

Felsenstein, J. (1985). *Evolution*, **39**, 783–791.

Feng, D.-F. & Doolittle, R. F. (1987). *J. Mol. Evol.* **25**, 351–360.

Fermi, G., Perutz, M. F., Shaanan, B. & Fourme, R. (1984). *J. Mol. Biol.* **175**, 159–174.

Fitch, W. M. & Margoliash, E. (1967). *Science*, **155**, 279–284.

Fujinaga, M. & James, M. N. G. (1987). *J. Mol. Biol.* **195**, 373–396.

Fujinaga, M., Dealbaere, L. T. J., Brayer, G. D. & James, M. N. G. (1985). *J. Mol. Biol.* **184**, 479–502.

Gotoh, O. (1982). *J. Mol. Biol.* **162**, 705–708.

Greer, J. (1981). *J. Mol. Biol.* **153**, 1027–1042.

Havel, T. F., Kuntz, I. D. & Crippen, G. M. (1983). *Bull. Math. Biol.* **45**, 665–720.

Hogeweg, P. & Hesper, B. (1984). *J. Mol. Evol.* **20**, 174–186.

Honzatko, R. B., Hendrickson, W. A. & Love, W. E. (1985). *J. Mol. Biol.* **184**, 147–164.

Hubbard, T. J. P. & Blundel, T. L. (1987). *Protein Engin.* **1**, 159–171.

James, M. N. G. & Sielecki, A. R. (1983). *J. Mol. Biol.* **163**, 299–361.

James, M. N. G., Delbaere, L. T. J. & Brayer, G. D. (1978). *Canad. J. Biochem.* **56**, 396–402.

James, M. N. G., Sielecki, A. R., Brayer, G. D., Delbaere, L. T. J. & Bauer, A. (1980). *J. Mol. Biol.* **144**, 43–88.

Johnson, M. S., Sutcliffe, M. J. & Blundell, T. L. (1989). *J. Mol. Evol.* **30**, 43–59.

Johnson, M. S., Šali, A. & Blundell, T. L. (1990). *Methods Enzymol.* In the press.

Jones, D. D. (1975). *J. Theoret. Biol.* **50**, 167–183.

Jones, T. H. & Thirup, S. (1986). *EMBO J.* **5**, 819–822.

Kabsch, W. & Sander, C. (1983). *Biopolymers*, **22**, 2577–2637.

KenKnight, C. E. (1984). *Acta Crystallogr. sect. A*, **40**, 708–712.

Kirkpatrick, S., Gelatt, C. D., Jr & Vecchi, M. P. (1983). *Science*, **220**, 671–680.

Lathrop, R. H., Webster, T. A. & Smith, T. F. (1987). *Commun. ACM*, **30**, 909–921.

Lesk, A. M. & Chothia, C. (1980). *J. Mol. Biol.* **136**, 225–270.

Lifson, D. J. & Sander, C. (1979). *Nature (London)*, **282**, 109–111.

Manavalan, P. & Ponnuswamy, P. K. (1978). *Nature (London)*, **275**, 673–674.

Matthews, B. W. & Rossmann, M. G. (1985). *Methods Enzymol.* **115**, 397–420.

Matthews, B. W., Grutter, M. G., Anderson, W. F. & Remington, S. J. (1981). *Nature (London)*, **290**, 334–335.

McGregor, M. J., Islam, S. A. & Sternberg, M. J. E. (1987). *J. Mol. Biol.* **198**, 295–310.

McLachlan, A. D. (1979). *J. Mol. Biol.* **128**, 49–79.

McLachlan, A. D. (1982). *Acta Crystallogr. sect. A*, **38**, 871–873.

McLachlan, A. D. & Shotton, D. M. (1971). *Nature (London)*, **229**, 202–205.

Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A. & Teller, E. (1953). *J. Chem. Phys.* **21**, 1087–1092.

Meyer, E., Cole, G., Radahakrishnan, R. & Epp, O. (1988). *Acta Crystallogr. sect. B*, **44**, 26–38.

Murthy, M. R. N. (1984). *FEBS Letters*, **168**, 97–102.

Needleman, S. B. & Wunsch, C. D. (1970). *J. Mol. Biol.* **48**, 443–453.

Overington, J. P., Sutcliffe, M. J., Watson, F., James, K., Campbell, S. & Blundell, T. L. (1988). In *Proceedings of an International Biotechnology Symposium 1* (Durand, G., Bobichon, L. & Florent, J., eds), pp. 279–304, Société Française de Microbiologie, Paris.

Palau, J., Argos, P. & Puigdomenech, P. (1982). *Int. J. Pept. Protein Res.* **19**, 394–401.

Pearl, L. H. & Taylor, W. R. (1987). *Nature (London)*, **329**, 351–354.

Press, W. H., Flannery, B. P., Teukolsky, S. A. & Vetterling, W. T. (1986). In *Numerical Recipes*, pp. 326–334, Cambridge University Press, Cambridge.

Rao, S. T. & Rossmann, M. G. (1973). *J. Mol. Biol.* **76**, 241–256.

Read, R. J. & James, M. N. G. (1988). *J. Mol. Biol.* **200**, 523–551.

Read, R. J., Fujinaga, A. R., Sielecki, A. R. & James, M. N. G. (1983). *Biochemistry*, **22**, 4420–4433.

Read, R. J., Brayer, G. D., Jurášek, L. & James, M. N. G. (1984). *Biochemistry*, **23**, 6570–6575.

Remington, S. J. & Matthews, B. W. (1978). *Proc. Nat. Acad. Sci., U.S.A.* **75**, 2180–2184.

Remington, S. J. & Matthews, B. W. (1980). *J. Mol. Biol.* **140**, 77–99.

Reynolds, R. A., Remington, J., Weaver, L. H., Fisher, R. G., Anderson, W. F., Ammon, H. L. & Matthews, B. W. (1985). *Acta Crystallogr. sect. B*, **41**, 139–147.

Richards, F. M. & Kundrot, C. E. (1988). *Proteins*, **3**, 71–84.

Richardson, J. S. (1977). *Nature (London)*, **268**, 495–500.

Richardson, J. S. (1981). *Advan. Protein Chem.* **64**, 167–339.

Richmond, T. J. & Richards, F. M. (1978). *J. Mol. Biol.* **119**, 537–555.

Rossmann, M. G. & Argos, P. (1975). *J. Biol. Chem.* **250**, 7525–7532.

Rossmann, M. G. & Argos, P. (1976). *J. Mol. Biol.* **105**, 75–95.

Rossmann, M. G. & Argos, P. (1977). *J. Mol. Biol.* **109**, 99–129.

Šali, A., Overington, J. P., Johnson, M. S. & Blundell, T. L. (1990). *Trends Biochem. Sci.*, in the press.

Sankoff, D. & Kruskal, J. B. (1983). *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, Addison-Wesley Publishing Company, Reading, MA.

Schulz, G. E. (1980). *J. Mol. Biol.* **138**, 335–347.

Sellers, P. H. (1979). *J. Comb. Theor. sect. A*, **16**, 253–258.

Sibanda, B. L. & Thornton, J. M. (1985). *Nature (London)*, **316**, 170–316.

Sibanda, B. L., Blundell, T. L. & Thornton, J. M. (1989). *J. Mol. Biol.* **206**, 759–778.

Steigemann, W. & Weber, E. (1979). *J. Mol. Biol.* **127**, 309–338.

Suguna, K., Bott, R. R., Padlan, E. A., Subramanian, E., Sheriff, S., Cohen, G. E. & Davies, D. R. (1987). *J. Mol. Biol.* **196**, 877–900.

Summers, N. L., Carson, W. D. & Karplus, M. (1987). *J. Mol. Biol.* **196**, 175–198.

Sutcliffe, M. J. (1988). Ph.D. thesis, University of London.

Sutcliffe, M. J., Haneef, I., Carney, D. & Blundell, T. L. (1987*a*). *Protein Engin.* **1**, 377–384.

Sutcliffe, M. J., Hayes, F. R. F. & Blundell, T. L. (1987*b*). *Protein Engin.* **1**, 385–392.

Takano, T. (1977). *J. Mol. Biol.* **110**, 569–584.

Tang, J., James, M. N. G., Hsu, I. N., Jenkins, J. A. & Blundell, T. L. (1978). *Nature (London)*, **217**, 618–621.

Taylor, W. R. (1986). *J. Theoret. Biol.* **119**, 205–218.

Tsukada, H. & Blow, D. M. (1985). *J. Mol. Biol.* **184**, 703–711.

Walter, J., Steigemann, W., Singh, T. P., Bartunik, H., Bode, W. & Huber, R. (1982). *Acta Crystallogr. sect. B*, **38**, 1462–1472.

Wistow, G., Turnell, B., Summers, L., Slingsby, C., Moss, D., Miller, L., Lindley, P. & Blundell, T. (1983). *J. Mol. Biol.* **170**, 175–202.

Edited by R. Huber