Gene, 73 (1988) 237–244
Elsevier

GEN 02726

CLUSTAL: a package for performing multiple sequence alignment on a microcomputer

(Cluster analysis; phylogenetic tree; protein secondary structure; RNA secondary structure; globin; 5S RNA; dendrogram)

Desmond G. Higgins and Paul M. Sharp

Department of Genetics, Trinity College, Dublin 2 (Ireland)

Received 17 July 1988
Accepted 4 August 1988
Received by publisher 23 August 1988

SUMMARY

An approach for performing multiple alignments of large numbers of amino acid or nucleotide sequences is described. The method is based on first deriving a phylogenetic tree from a matrix of all pairwise sequence similarity scores, obtained using a fast pairwise alignment algorithm. Then the multiple alignment is achieved from a series of pairwise alignments of clusters of sequences, following the order of branching in the tree. The method is sufficiently fast and economical with memory to be easily implemented on a microcomputer, and yet the results obtained are comparable to those from packages requiring mainframe computer facilities.

INTRODUCTION

The recent great increase in the extent of biological macromolecule sequence determination has meant that the simultaneous alignment of three or more nucleic acid or amino acid sequences has become a common necessity. Simply, the procedure involves the insertion of gaps in the sequences so as to maximise the overall similarity. Such 'multiple alignments' are useful first in demonstrating similarity among members of a sequence family. When used as part of a database search, multiple alignments have the advantage of emphasizing conserved regions, allowing the detection of distantly related proteins (Gribskow et al., 1987; Patthy, 1987). The alignment may then be used to predict protein secondary structures, if the crystal structure of one or more members of the family have already been determined (e.g., Barton and Sternberg, 1987). The alignment may also serve as a prelude to inferring patterns of mutational change and/or the degree of evolutionary relationship among family members, often leading ultimately to the reconstruction of an evolutionary tree (e.g., Feng and Doolittle, 1987).

For pairwise alignment (i.e., only two sequences) the dynamic programming algorithm of Needleman and Wunsch (1970) has been extensively used, since it is guaranteed to maximise the overall similarity between the two sequences for any given gap penalty. However, this and other similar exact methods have heavy computer time and space (core memory)

Correspondence to: Dr. D.G. Higgins, Department of Genetics, Trinity College, Dublin 2 (Ireland) Tel. (353) 1-772941.

Abbreviations: aa, amino acid(s); ASCII, American Standard Code for Information Interchange; nt, nucleotide(s); S, sedimentation constant; UPGMA, Unweighted Pair-Group Method using Arithmetic Averages.

requirements, so that approximate, heuristic approaches, such as that of Wilbur and Lipman (1984), have become popular, particularly when a large number of pairwise comparisons are being made. Such methods are fast, and yield virtually the same results as the exact methods, as long as the sequences are not too dissimilar. For exact multiple alignment procedures the problems with computer time and memory increase approximately exponentially with the number of sequences compared, so that all algorithms described for more than three sequences have been heuristic (e.g., Bains, 1986; Santibanez and Rohde, 1987; Sobel and Martinez, 1986; Waterman, 1986). Even so, most of these methods utilise minicomputers or mainframes.

Here we describe a fast method for obtaining multiple alignments of a large number of nucleotide or amino acid sequences on a microcomputer. The method is in essence a 'quick and dirty' version of the algorithm of Feng and Doolittle (1987). The rationale is to first find a phylogenetic tree for the sequences derived from a matrix of all pairwise sequence similarities, and then to progressively align the most similar sequences, substituting a consensus for each pair as they are aligned, with gaps that occur in early alignments being preserved through the later stages. We judge the results to be good, in so far as the alignments we derive differ very little from those obtained using algorithms requiring mainframe computer facilities. The programme, named CLUSTAL, is available upon request.

MATERIALS AND METHODS

(a) System

The package consists of three main programmes plus some utilities. All code was written in standard FORTRAN 77 and compiled for microcomputers using the Microsoft FORTRAN compiler version 4.0. Program performance was tested on a IBM AT compatible microcomputer, running at 10 MHz with no coprocessor, 640 kilobytes of memory and a hard disk.

Copies of the executable files, documentation and test data files will be sent on request. Please send three 5.25-inch floppies formatted to 360 kilobytes or one high-density floppy formatted at 1.2 megabytes. A hard disk is *strongly* recommended for these programs. No special graphics equipment is required.

(b) Algorithms

There are three distinct stages in the multiple alignment strategy described here (see Fig. 1 for a flowchart). The three stages are: (*1*) calculation of all pairwise sequence similarities; (*2*) construction of a dendrogram from the similarity matrix generated in stage 1; (*3*) multiple alignment of the sequences in a pairwise manner, following the order of clustering in the dendrogram from stage 2. To maintain flexibility, the three stages are maintained as three separate programs which interact by way of intermediate text files. This is desirable because it offers the user the opportunity to intercept intermediate results for analysis by other software. Further, any of the stages may be changed independently of the other two, for

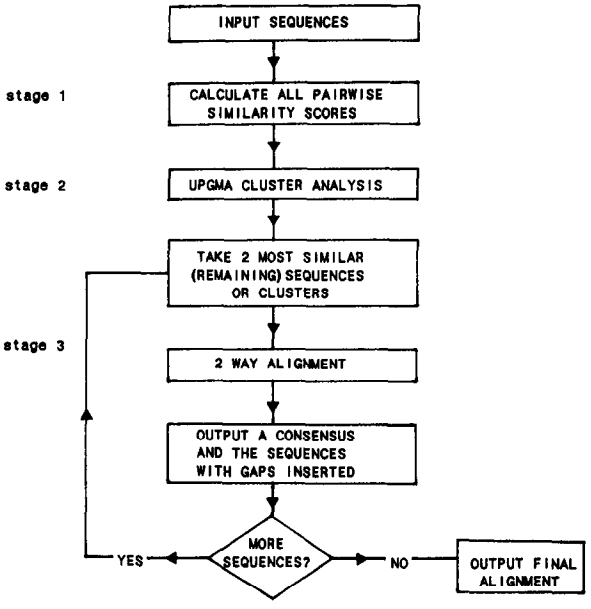

Fig. 1. Flow chart of the multiple alignment strategy described in MATERIALS AND METHODS, section **b**. The core of the multiple alignment process takes place in stage three. At each step, in stage three, two clusters consisting of one or more sequences each are aligned. After each alignment a consensus, which will be used to represent the two aligned clusters in future alignments, is stored and gaps are inserted in the original sequences at appropriate positions. When all sequences have been aligned, the process is complete and the full alignment is outputted.

example, to make use of alternative algorithms for the estimation of pairwise similarities, or for the cluster analysis.

(1) Pairwise similarities

The number of scores required to fill out a similarity matrix is $0.5\ N\ (N-1)$ where N is the number of sequences. For large N this involves the calculation of a considerable number of scores, e.g., the number of values is 4950 when $N = 100$. Therefore a fast method for calculating the similarities is needed. The algorithm used is that of Wilbur and Lipman (1984). The scores are calculated as the number of exactly matching residues between two sequences in the optimal alignment, minus a fixed penalty for every gap. Optionally, only residues that are part of matches of a given length (k-tuple matches) are scored, e.g., two aa residues for proteins, or 2 to 4 nt for nucleic acid sequences. The combination of fixed gap penalties and k-tuple matches allows the calculation of several scores per second for similar and/or short sequences on a microcomputer. Importantly, no attempt is made to assess the significance of scores; relative magnitude is all that is needed for the next stage, i.e., construction of the dendrogram. Input to this program can be either a text file containing a series of file names (one sequence per file) or as a single file with a right angle bracket ('>') delimiting the start of each sequence.

(2) Cluster analysis

Numerous methods for performing cluster analysis or deriving hypothetical phylogenetic trees from distance/similarity matrices have been proposed (see Sneath and Sokal, 1973, or Nei, 1987, for reviews). The basic criterion in choosing a method here is that it should be able to handle very large matrices. The UPGMA method (Sneath and Sokal, 1973) was chosen as it is in widespread use, is conceptually simple and has modest time and space requirements. The program used here accepts as input a file of similarity scores and outputs a dendrogram as a series of records, one for each cluster in the analysis. This dendrogram file is used as input to the alignment program. At this stage, the dendrogram is of interest in its own right and can be used as input to a utility program for display on the screen. The time required to construct a dendrogram for 100 sequences is approximately 3 min on a microcomputer.

(3) Multiple alignment

This is the core of the package. The dendrogram file from stage two and the original sequence data file(s) are used as input. The sequences are taken, and aligned using the Wilbur and Lipman (1984) method, following the order of clustering in the dendrogram. After each alignment, the aligned sequences with gaps inserted at appropriate positions (dashes) and a consensus sequence are written to a work file (random access, unformatted records). In the simplest case, the consensus is an exact consensus, i.e., only residues found at a given position in all sequences are stored, otherwise an unknown residue ('X') is recorded. At no stage are all of the original or consensus sequences stored in memory. As sequences are aligned the consensus is used to represent the cluster in later alignments. Thus, when two clusters are joined together this is achieved by reading the consensus sequences from the work file and aligning these. This simple approach will work for closely related sequences but there is an absolute requirement for at least a small percentage of all the residues to be conserved across all sequences. However, two modifications are used to dramatically increase sensitivity; these allow for (*i*) conservative substitutions in protein alignments; and (*ii*) a small degree of mismatch in the consensus sequences.

(*i*) *Conservative substitutions*. A four-tier weighting scheme, based on the log-odds matrix of Dayhoff (1978), was used to differentially weight aligned residues in protein alignments (Fig. 2). Using a cut-off point for deciding whether or not a substitution is conservative (the cut-off is set by the user at run time; 10 is the default), the four classes of match and their weights are:

(I) unmatched residues with a Dayhoff (1978) score of < 'cut-off' : score 0
(II) unmatched residues with a Dayhoff (1978) score > or = 'cut-off' : score 1
(III) exactly matched residues (except Cys, Phe, Trp or Tyr) : score 2
(IV) exactly matched Cys, Phe, Trp or Tyr : score 3

This simple four-tier system slows down the two-way alignment process and increases memory requirements by a factor of 2 when a cut-off of 10 is used, compared to the usual implementation of the Wilbur and Lipman (1984) algorithm. Because of the speed of the method in the first place, this is acceptable.

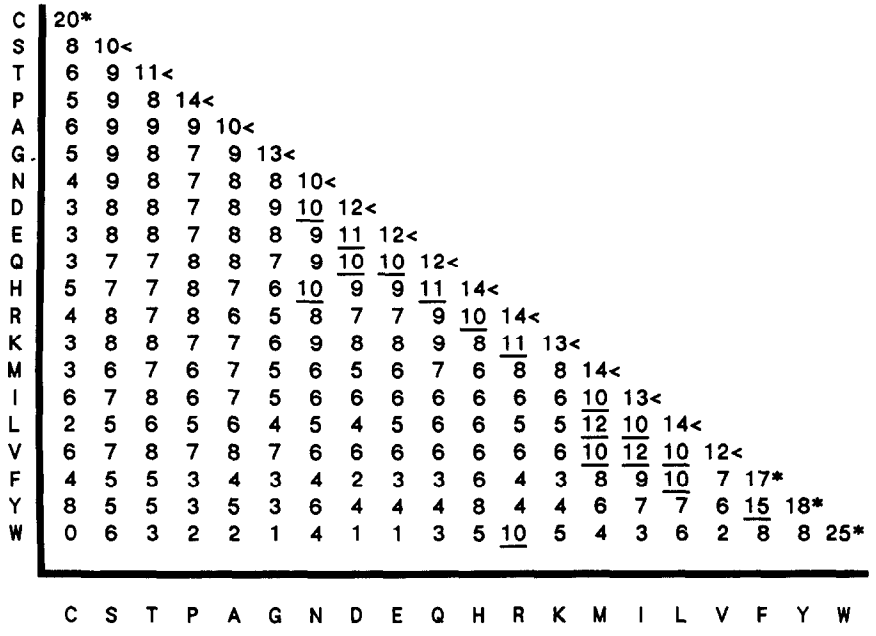

Fig. 2. Log-odds amino acid similarity matrix of Dayhoff (1978). The values in the matrix (scaled to lie between 0 and 25) indicate the frequency of substitution of any amino acid by another in related sequences. The single-letter amino acid code is used. The four-tier weighting system described in MATERIALS AND METHODS, section *b3i* (conservative substitutions), is illustrated. When a cut-off of 10 is used to decide whether or not a replacement is conservative, the four weights are indicated by: N (score = 0); $\underline{N}$ (score = 1); $N<$ (score = 2); N^* (score = 3), where N is a value from the matrix.

The effect on alignment sensitivity is dramatic in the case of highly diverged proteins. In the usual implementation of the Wilbur and Lipman (1984) method, the speed and economy with memory is largely achieved by maintaining look-up tables (Dumas and Ninio, 1982) of exactly matching fragments (k-tuple matches) between sequences, generated from single passes through each sequence. This approach is not easily compatible with a full Dayhoff matrix scoring system. The four-tier system, described above, however, is easily incorporated by chaining entries in the look-up tables corresponding to conservative substitutions.

(*ii*) *Partial consensus sequences.* Ideally, one would like a consensus sequence to contain information on the variability of residues at each position. The alignment strategy followed here prohibits the use of most of this information when two clusters of sequences are aligned. Nonetheless, it is important that at least a small degree of mismatch is allowed at any position in a consensus. The approach used is to include a residue in the consensus if it occurs in more than 75% of the sequences that the consensus represents, otherwise it is recorded as an 'unknown' residue. This is achieved by manipulating the ASCII values of characters in the consensus, to record whether or not any mismatches have occurred up to any particular alignment stage, rather than counting the occurrences of each residue at every position after each alignment.

RESULTS AND DISCUSSION

(a) Alignments

Two examples of alignments carried out using CLUSTAL are described below. These sets of sequences were chosen because, in both cases, there are some criteria for judging the quality of the alignments, other than just the magnitude of an overall similarity score. The examples are (*1*) seven divergent members of the globin family, and (*2*) 34 5S ribosomal RNA sequences.

(1) Globins

Lesk and Chothia (1980) identified seven α-helices, homologous between seven globin sequences, whose crystal structures had been deter-

mined and Barton and Sternberg (1987) have used these sequences as a test of the accuracy of their multiple alignment strategy. Barton and Sternberg found that, although the sequences are highly diverged in terms of primary structure, they could correctly align all except two of the residues in each α-helix, across all the sequences. The samples includes two mammalian α-globins, two β-globins, a myoglobin, a cyanohaemoglobin and a leghaemoglobin. CLUSTAL yields the alignment shown in Fig. 3. The α-helices are labelled A to H and, as can be seen, most of them are correctly aligned. Only eight residues are exactly conserved in all of the molecules but the method still succeeds in aligning the regions of homologous secondary structure more or less correctly. Barton and Sternberg (1987) found the same eight exactly conserved residues, but their method was slightly more accurate in aligning the α-helices.

(2) 5S RNAs

Hori et al. (1985) produced an alignment of 34 5S RNA sequences taken mainly from plants but including sequences from yeast, bacteria and a chloroplast. The alignment of Hori et al. (1985) was produced by manually aligning regions of known secondary structure (loops and base-paired regions in stems) followed by 'eyeball adjustment'. We tested the sensitivity of our package using the same sequences. Our automatic alignment succeeds in consistently aligning the regions of homologous secondary structure and produces an alignment that is not very different from that of Hori et al. (1985).

(b) Dendrograms

The dendrograms, used as input to the final alignment stage, are not intended to be used as phylogenetic trees. There are two reasons for this. First, Wilbur and Lipman (1984) alignment scores are only crude estimates of sequence similarity. No corrections are made for multiple substitutions or replacements. Further, when filtering is used to find diagonals with large numbers of k-tuple matches, the resulting scores are not metric quantities (Wilbur and Lipman, 1984). Thus the branch lengths will not be to scale. Second, UPGMA has been criticised as a method for inferring phylogenies from sequence data due to its inability to deal with unequal rates of evolution along different lineages (e.g., Nei, 1987). This may lead to errors in the tree topology (branching order). Nonetheless, it is important for the success of the alignment strategy outlined here that the dendrograms used have biologically plausible topologies, since the order of clustering does have an effect on the position of gaps in the final alignment. The dendrogram obtained by CLUSTAL during the course of aligning the 5S RNA sequences is shown in Fig. 4. As can be seen, the sequences corresponding to the major plant groups (Bryophytes, Pteridophytes, Gymnosperms, etc.) do cluster in an appropriate manner. Indeed, the topology is almost

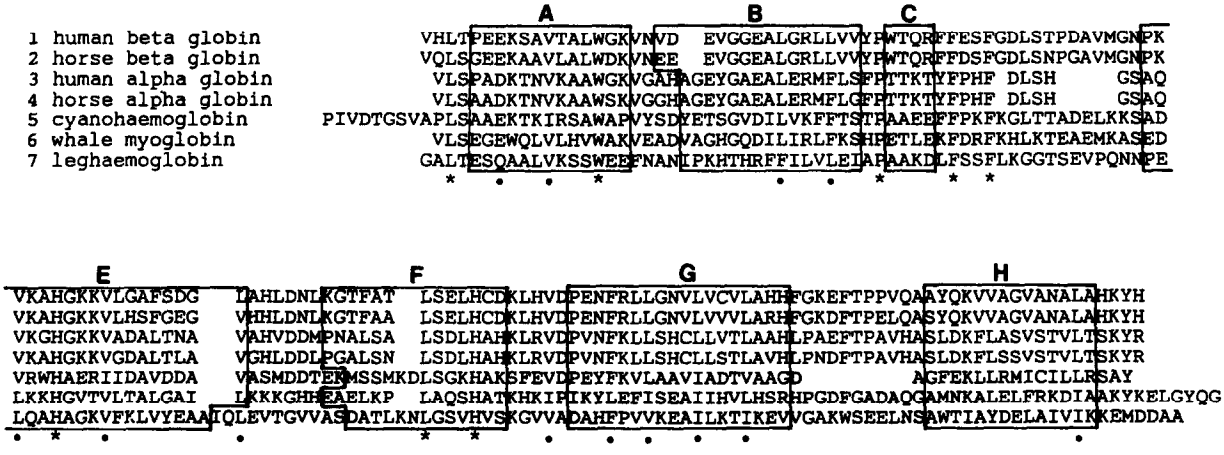

Fig. 3. CLUSTAL-produced multiple alignment of seven globin sequences taken from Lesk and Chothia (1980) (see RESULTS, section *a1*). Seven α-helices, homologous between all sequences, are labelled A to H and boxed. Asterisks indicate residues exactly conserved across all the sequences, while dots indicate regions where all pairs of residue have a similarity score greater than or equal to 10 (see Fig. 2).

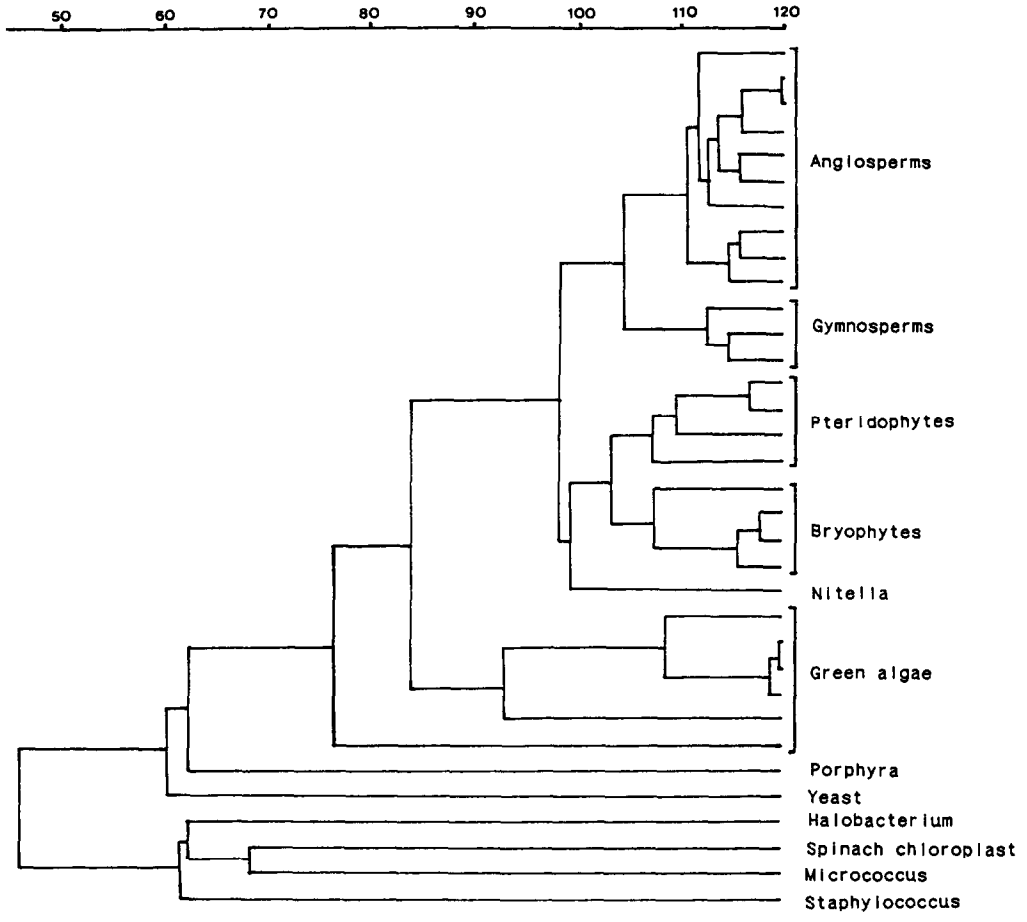

Fig. 4. UPGMA dendrogram of 34 plant, yeast and bacterial 5S RNA sequences. The sequences were taken from Hori et al. (1985) (see RESULTS, section a2). The major plant taxonomic groupings are indicated. The scale across the top margin shows the number of matching nucleotides (after alignment) between two clusters or sequences.

identical to that produced by Hori et al. (1985) using a more sophisticated strategy. In general, we have found that CLUSTAL yields a satisfactory topology in a wide variety of situations. In cases where this is not so, the user is free to manually adjust the topology or use an alternative approach.

(c) Execution speed

The exact speed of the multiple alignment process will depend on the choice of parameters used to control the alignment, and on the length and number of sequences. Both of the examples given above were carried out in less than 5 min on a microcomputer. For large numbers of sequences, by far the greatest fraction of the time required for a multiple alignment is taken up in calculating the pairwise sequence similarities to construct the dendrogram. For example, using 92 cytochrome *c* sequences, calculation of the similarity matrix and construction of the dendrogram requires 25 min. Once constructed, the same dendrogram can be used repeatedly as input to the alignment stage which typically requires between 4 and 10 min for the 92 cytochromes, depending on the choice of parameters.

(d) Comparison with other methods

Over the past three years, more than a dozen methods for performing multiple sequence alignments have been published. The method of Murata et al. (1985) is an extension for three sequences of the dynamic programming pairwise-alignment algorithm of Needleman and Wunsch (1970). It gives an exactly optimal solution to the alignment problem but has heavy computing requirements. Indeed, an extension

of the method to four or more sequences would be impossible with currently available computer hardware, except for very short sequences. Therefore, all of the methods described for aligning many sequences have been heuristic. Some methods are based on iteratively building a consensus sequence of all of the sequences to be aligned (e.g., Bains, 1986). Other methods are based on finding sub-sequences in common between all or some of the sequences (e.g., Sobel and Martinez, 1986; Waterman, 1986). Santibanez and Rohde (1987) describe an extension of the fast two-sequence method of Wilbur and Lipman (1984) to three sequences.

The present approach belongs to a family of methods that work by a series of pairwise alignments. In the method of Feng and Doolittle (1987) and in the method described here, the sequences are progressively aligned according to the order of branching in a hypothetical phylogenetic tree. Taylor (1987) and Barton and Sternberg (1987) use ordered lists of sequences, derived from pairwise sequence distances to determine the order of alignment. All of these pairwise methods have the advantage of being very fast and economical with computer memory, allowing the alignment of large numbers of sequences. Placing the gaps in a progressive manner, rather than trying to do so for all sequences simultaneously, enormously reduces the complexity of the problem. Speed however, is not the only, or indeed the most important criterion for judging the usefulness of a multiple alignment method. The most important criterion is whether or not the alignments are useful for evolutionary studies or in correctly aligning homologous secondary structure features.

Feng and Doolittle (1987) justify a progressive approach to multiple alignment by arguing that greater weight should be attached to gaps that result from the comparison of closely related sequences. These gaps should not be changed because of minor improvements in the alignment when more distantly related sequences are included. They found that phylogenetic trees based on distance matrices, derived from multiply aligned sequences, often have more plausible topologies than trees based on distances derived from separate pairwise alignments.

Barton and Sternberg (1987) justify their approach because of the accuracy with which regions of known protein secondary structure can be aligned in sets of sequences where crystal structures are known. Just as Feng and Doolittle (1987) found an improvement in tree topology after multiple alignment, Barton and Sternberg found that secondary structure regions could be more accurately aligned when several sequences are aligned together, than with separate pairwise alignments. Taylor (1987) compared the results of using his method to align three copper-binding proteins with an exact solution obtained by Murata et al. (1985). While the two alignments were basically similar, there were some obvious differences. Taylor (1987) argued that these differences occurred in regions where the sequences had diverged to such an extent that no particular arrangement could be considered 'better' than another, biologically.

The main advantages of the multiple alignment strategy described here are its simplicity, speed and flexibility. For similar sequences, the approach will find a solution that is difficult to improve by 'eyeball manipulation'. For more divergent sequences, it can be used to find a good initial approximation to an exact alignment. In situations such as evolutionary studies, where precise alignment is desired, the exactly optimal solution may be unobtainable anyway. One is then left with a choice of using one of the better heuristic methods (still not guaranteed to be optimal) or resigning oneself to manual alignment.

ACKNOWLEDGEMENTS

This is a publication from the Irish National Centre for Bioinformatics. This work was supported by grant No. BAP-0137-IRL from the European Community Biotechnology Action Programme.

REFERENCES

Bains, W.: MULTAN: a program to align multiple DNA sequences. Nucleic Acids Res. 14 (1986) 159–177.

Barton, J.B. and Sternberg, M.J.E.: A strategy for the rapid multiple alignment of protein sequences. J. Mol. Biol. 198 (1987) 327–337.

Dayhoff, M.O.: A model of evolutionary change in proteins. Matrices for detecting distant relationships. In Dayhoff, M.O. (Ed.), Atlas of Protein Sequence and Structure. Vol. 5, Suppl. 3. National Biomedical Research Foundation, Washington DC, 1978, pp. 345–358.

Dumas, J.-P. and Ninio, J.: Efficient algorithms for folding and comparing nucleic acid sequences. Nucleic Acids Res. 10 (1982) 197–206.

Feng, D.-F. and Doolittle, R.F.: Progressive sequence alignment as a prerequisite to correct phylogenetic trees. J. Mol. Evol. 25 (1987) 351–360.

Gribskow, M., McLachlan, A.D. and Eisenberg, D.: Profile analysis: detection of distantly related proteins. Proc. Natl. Acad. Sci. USA 84 (1987) 4355–4358.

Hori, H., Lim, B.-L., Ohama, T., Kumazaki, T. and Osawa, S.: Evolution of organisms deduced from 5S rRNA sequences. In Ohta, T. and Aoki, K. (Ed.), Population Genetics and Molecular Evolution. Japan Science Society Press, Tokyo, 1985, pp. 369–384.

Lesk, A.M. and Chothia, C.: How different amino acid sequences determine similar protein structures: the structure and evolutionary dynamics of the globins. J. Mol. Biol. 136 (1980) 225–270.

Murata, M., Richardson, J.S. and Sussman, J.L.: Simultaneous comparison of three protein sequences. Proc. Natl. Acad. Sci. USA 82 (1985) 3073–3077.

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

Nei, M.: Molecular Evolutionary Genetics. Columbia University Press, New York, 1987.

Patthy, L.: Detecting homology of distantly related proteins with consensus sequences. J. Mol. Biol. 198 (1987) 567–577.

Santibanez, M. and Rohde, K.: A multiple alignment program for protein sequences. Comp. Appl. Biosci. 3 (1987) 111–114.

Sneath, P.H.A. and Sokal, R.R.: Numerical Taxonomy. Freeman, San Francisco, 1973.

Sobel, E. and Martinez, H.M.: A multiple sequence alignment program. Nucleic Acids Res. 14 (1986) 363–374.

Taylor, W.R.: Multiple sequence alignment by a pairwise algorithm. Comp. Appl. Biosci. 3 (1987) 81–87.

Waterman, M.S.: Multiple sequence alignment by consensus. Nucleic Acids Res. 14 (1986) 9095–9102.

Wilbur, W.J. and Lipman, D.J.: The context dependent comparison of biological sequences. SIAM J. Appl. Math. 44 (1984) 557–567.

Communicated by K.F. Chater.