Volume 16 Number 22 1988 Nucleic Acids Research

Multiple sequence alignment with hierarchical clustering

Florence Corpet

Laboratoire de Génétique Cellulaire, INRA Toulouse, BP 27, 31326 Castanet Tolosan, France

Received July 7, 1988; Revised and Accepted October 14, 1988

ABSTRACT

An algorithm is presented for the multiple alignment of sequences, either proteins or nucleic acids, that is both accurate and easy to use on microcomputers. The approach is based on the conventional dynamic-programming method of pairwise alignment. Initially, a hierarchical clustering of the sequences is performed using the matrix of the pairwise alignment scores. The closest sequences are aligned creating groups of aligned sequences. Then close groups are aligned until all sequences are aligned in one group. The pairwise alignments included in the multiple alignment form a new matrix that is used to produce a hierarchical clustering. If it is different from the first one, iteration of the process can be performed. The method is illustrated by an example : a global alignment of 39 sequences of cytochrome c.

INTRODUCTION

Macromolecules, either nucleic acids or proteins, are sequenced by an increasing number of molecular biologists, by using techniques which are automatized and fast. Hence an increasing bulk of sequences has to be analyzed, which is impossible without the help of data processors.

It is of particular interest to locate the parts that are common to many sequences in a same family. For example, the homologous regions in sequence of proteins having the same activities in various living organisms, are likely to be the most important from functional and structural points of view.

This kind of analysis requires the alignment of the sequences, *id est* a representation of the sequences in rows, the homologous regions being in the same columns. When one knows the 3D structures of the macromolecules, this problem is easy to solve. But this is a very rare case and biologists want to align macromolecules when they know only the sequences.

The alignment of two sequences can be performed with any of several programs that have been developed since 1970 (1). However, when there are more than two sequences, the result of pairwise comparisons may not be consistent for certain residues. But simultaneous comparison of all the sequences

cannot, in practice, be executed since the number of segment comparisons that must be carried out is of the order of the product of the sequence lengths.

Hence, strategies have been chosen to propose alternative approaches : limiting the problem to three short sequences (2) or to closely related sequences (3), using a predetermined evolutionary tree (4), finding common subsequences (5 to 8), selecting the best pairwise alignments from the scores of all pairwise comparisons (9). Several multiple sequence alignment programs utilize the technics of pairwise alignments, building the final alignment by gradually aligning further sequences, according to the basic Needleman-Wunsch procedure (1). Based upon the scores of the initial alignments of all pairs of sequences, different strategies are used to determine in what order the sequences are incorporated into the final alignment. Since the sequences in the already aligned set preserve their relative structure, finding the proper order is crucial. Most algorithms proceed in a sequential way (10 to 12). Feng and Doolittle utilize a more sophisticated clustering, and the closely related subsets of sequences are prealigned before the final alignment (13).

The algorithm described in this paper also uses clustering but in a more simple way and there is no prealignment of clusters.

DESCRIPTION OF THE METHOD

Algorithm for two sequences

Let the two sequences A and B have lengths m and n, and denote by A(i) and B(j) the i-th and j-th elements in the respective sequences. To every pair of elements A(i), B(j), a weight w(i,j) is assigned from a suitable matrix D such as the mutation data matrix of Dayhoff (14) for amino acids (if necessary a suitable constant must be added to make all matrix entries non-negative) : $w(i,j) = D(A(i),B(j))$. The values of w are not stored but computed when needed, from stored values of the matrix. The method of computation used, following that of Needleman and Wunsh (1) and of Murata et al.(2), is to work backward from the cell (m,n), calculating the maximum total value S for paths from each cell. Let S(i,j) be the maximum, over all paths from cell (i,j) to the bottom or the right side, of the sums of values w over the cells in the path minus g times the number of gaps in the path. This gap penalty is independent of the gap length, as suggested by the results of Barton and Sternberg (15). Let M(i,j) be the maximum value of S over all the cells (i,k) and (l,j) where $j \leq k \leq n$ and $i \leq l \leq m$ (as a consequence of its definition, M(i,j) is also the maximum value of S over all cells (l,k) with $l \geq i$ and $k \geq j$).

The following algorithm is used to calculate S and M :

$$S(i,j) = w(i,j) + \max (S(i+1,j+1), M(i+1,j+1) - g);$$
$$M(i,j) = \max (S(i,j), M(i+1,j), M(i,j+1)).$$

Once the matrix S has been calculated, a traceback procedure is performed to find the successive cells of the best path. Its first cell is the one with the maximum value in the first row or the first column. This value is the score of the alignment. No gap penalty is added at either end of the path.

Alignment of two clusters of aligned sequences

To compare more than two sequences, clusters of aligned sequences are regrouped step by step with an alignment algorithm, which is an extension of the preceding one (10).

Let B_1 ... B_P be the sequences of one cluster and $C_1 \ldots C_Q$ be the ones of a second cluster. When generating the matrix S in order to align sequences $C_1 \ldots C_Q$ with sequences $B_1 \ldots B_P$, a scoring scheme is adopted that includes a contribution from all previously aligned sequences, thus giving more weight to regions already aligned. Let i (resp. j) be the position of an aligned residue in sequences $B_1 \ldots B_P$ (resp. $C_1 \ldots C_Q$), then :

$$w(i,j) := \frac{1}{P.Q} \sum_{R=1}^{R=P} \sum_{S=1}^{S=Q} D(B_R(i), C_S(j)).$$

where D is the matrix of amino acid pair scores.

For example, the weight of (Ala-Val-Leu) aligned with (Ala-Leu) is given by the weight of [(Ala *vs.* Ala) + (Ala *vs.* Leu) + (Val *vs.* Ala) + (Val *vs.* Leu) + (Leu *vs.* Ala) + (Leu *vs.* Leu)] x 1/6. The value of D when one of its indices is a gap is set to 0.

Matrices S and M are computed as before with the new values of w. One obtains a cluster of P + Q aligned sequences that takes the place of the two clusters of P and Q aligned sequences.

Order of clustering

The order in which the sequences are compared can have an effect on the multiple final alignment, hence a good order must be chosen. The method used here is to perform a hierarchical clustering of the sequences, using the scores of the pairwise comparisons as an index of similarity between sequences. The principle is simple : the hierarchy is built from its base (the set of sequences)

creating new clusters by union of two already created clusters that are the closest ones (16).

Let A_1, A_2 ... A_N be the N sequences to be aligned. All pairwise comparisons are performed by a fast algorithm (17) and stored in a matrix T_1 : $T_1(I,J)$ is the score of the alignment of A_I with A_J. Then clusters of aligned sequences are defined as follows. At step 1, there are N clusters, every one including one sequence. The best score $T_1(I,J)$ is found in the matrix T_1. The sequences A_I and A_J (whose score is the best) are aligned and the alignment of the two sequences form a cluster that takes the place of the I-th sequence. The J-th sequence is suppressed. A new matrix of score T_2 is built ; its dimension is (N-1) and it is a copy of T_1 where column and row J are deleted and column and row I are built from columns and rows I and J of T_1 : $T_2(I,K)$ is the mean of the two scores $T_1(I,K)$ and $T_1(J,K)$. $T_2(I,K)$ is called the "score" of cluster I *vs.* cluster K. At step s (s = 1,2,... N-1), there are N - s + 1 clusters of sequences and the matrix T_S holds the "scores" between these clusters. If the greatest element of T_S is $T_S(I,J)$, clusters I and J are aligned and the resulting alignment forms the new cluster I. Cluster J is deleted. The matrix T_{S+1} is built as follows :

$$T_{S+1}(K,L) = T_S(K,L) \quad \text{if} \quad K,L \neq I,J ,$$

$T_{S+1}(J,K)$ and $T_{S+1}(K,J)$ does not exist for any K,

$$T_{S+1}(I,K) = T_S(K,I) = \frac{N_I . T_S(I,K) + N_J . T_S(J,K)}{N_I + N_J} \quad \text{if } K \neq I,J$$

where N_I (resp. N_J) is the number of sequences in the cluster I (resp. J).

At every step, the number of clusters decreases by one, and one of the new clusters includes the sequences of two clusters of the preceding step. At step N, there is one cluster including the N aligned sequences.

Presentation of the complete process

0 - initialization : all pairwise comparisons are performed by a fast algorithm (17) and their scores are recorded.

1 - a hierarchical clustering of the sequences is done using these scores.

2 - the hierarchical tree is climbed with the pairwise alignment of clusters to obtain the complete alignment.

3 - the alignment is shown, recorded or printed. A score is given for the multiple alignment : it is the sum of the scores of all the pairwise alignments included in the multiple one.

4 - A new hierarchical clustering is done with these new scores.

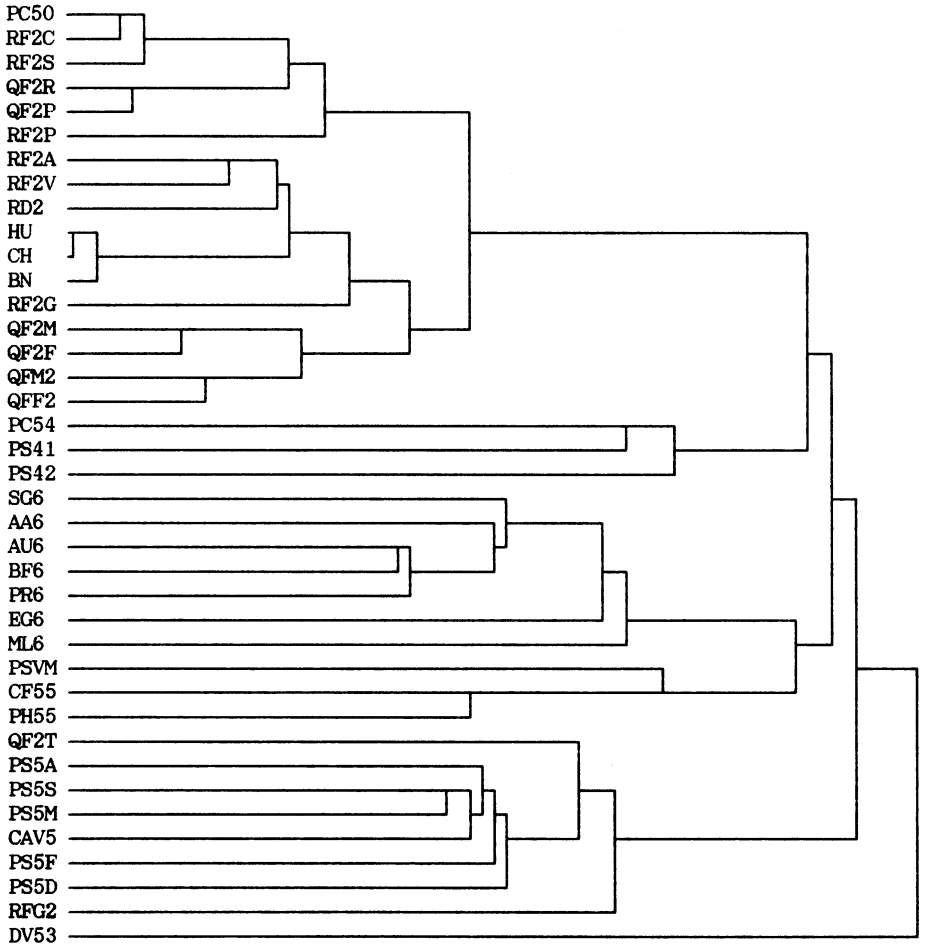

Figure 1 - Hierarchical clustering of 39 related bacterial, algal and mitochondrial cytochrome c. PC50: Paracoccus denitrificans c_{550}, RF2S: Rhodopseudomonas sphaeroides c_2, RF2C: R. capsulata c_2, RF2P: R. palustris c_2, RF2A: R. acidophila c_2, RF2V: R. viridis c_2, RF2G: R. globiformis c_2, RFG2: R. gelatinosa c_2, HU: Human c, CH: Chicken c, BN: Tuna c, RD2: Rhodomicrobium vanielii c_2, QF2R: Rhodospirillum rubrum c_2, QF2P: R. photometricum c_2, QF2M: R. molischianum c_2, iso-1, QF2F: R. fulvum c_2, iso-1, QFM2: R. molischianum c_2, iso-2, QFF2: R. fulvum c_2, iso-2, QF2T: R. tenue c_2, PS5A: Pseudomonas aeruginosa c_{551}, PS5F: P. fluorescens biotype c_{551}, PS5S: P. stutzeri c_{551}, PS5M: P. mendocina c_{551}, PS5D: P. denitrificans c_{551}, PS41: P. aeruginosa c_4, 1st half, PS42: P. aeruginosa c_4, 2nd half, PSVM: P. mendocina c_5, AV5: Azotobacter vinelandii c_{551}, SG6: Spirulina maxima c_6, AA6: Anacystis nidulans c_6, AU6: Alaria esculenta c_6, PR6: Porphyra tenera c_6, BF6: Bumilleriopsis filiformis c_6, ML6: Monochrysis lutheri c_6, EG6: Euglena gracilis c_6, PC54: Paracoccus sp. c_{554}, CF55: Chlorobium limicola f.sp. c_{555}, PH55: Prosthecochloris aestuarii c_{555}, DV53: Desulfovibrio vulgaris c_{553}.

```
      1         10          20          30          40          50          60
PC50 QDGDAAkGEKeFn.K.CkaCHmlqapdGTDIl.KGgKtGPNLYGVVGRkiaSeegFk.YgEgilEVaeknpd
RF2S QEGDpeaGaKaFn.q.CqtCHvIvddsGTtIagRnAKtGPNLYGVVGRTAgTqadFkgYgEgmkEaGA..kG
RF2C  GDAAkGEKeFn.K.CktCHsIiapdGTEIV.KGAKtGPNLYGVVGRTAgTypeFk.YkDsivaLGA..sG
QF2R  EGDAAaGEKvsk.K.ClaCHtfdqggan.......KvGPNLFGVfenTAahkdnYa.YsEsytEMkA..kG
QF2P  aGDAAvGEKiakaK.CtaCHdLnkggpi.......KvGPpLFGVfGRTtgTfagYs.YspgytvMGq..kG
RF2P   QDAAkGEavFk.q.CmtCHradkn.........mvGPaLgGVVGRkAgTaagFt.YsplnhNsGe..aG

RF2A  AGDpdaGqkvFlk..CaACHkIgPgaKN.......GVGPSLnGVanRKaGqaeGFa.YSDAnkn.....SG
RF2V   qDaAsGeqvFkQ..ClvCHSIgPgaKN.......kVGPvLnGLFGRHsGtieGFs.YSDAnkn.....SG
HU     GDvekGkkiFimk.CsqCHTVEkgGKh.......ktGPnLhGLFGRKtGqapGYs.YTaAnkn.....kG
CH     GDiekGkkiFvQk.CsqCHTVEkgGKh.......ktGPnLhGLFGRKtGqaeGFs.YTDAnkn.....kG
BN     GDvAkGkktFvQk.CaqCHTVEngGKh.......kVGPnLwGLFGRKtGqaeGYs.YTDAnks.....kG
RD2   AGDpvkGeqvFkQ..CKiCHqVgPtaKN.......GVGPeqndVFGqKaGarpGFn.YSDAmkn.....SG
QF2M     adapppaFnQ..CKACHSID.aGKN.......GVGPSLsGaYGRKvGlapnYk.YSpAhla.....SG
QF2F      AdaptaFnQ..CKACHSIE.aGKN.......GVGPSLsGaYGRKvGlapnYk.YSaAhla.....SG
QFM2      AdapagFtl..CKACHSVE.aGKN.......GVGPSLaGVYGRKaGtisGFk.FSDphik.....SG
QFF2      AdappaFgm..CKACHSVE.aGKN.......GVGPSLaGVYGRKaGtlaGFk.FSDphak.....SG
RF2G[1] ppGDpveGkhlFhti.CilCHT.DikGrN.......kVGPSLyGVvGRHsGiepGYn.YSEAnik.....SG

QF2T       adesaLaqTKgClACHnpEkKV.........VGPAYgwVAkKYAGQaGA.................
RFG2       atpaeLatkagCavCHqptaKg.........LGPsYqEIAkKYkGQaGA.................
PS5A       edpEvLFKnKgCvACHaIDtKM.........VGPAYKDVAAKFAGQaGA.................
PS5F       edGaaLFKSKpCaACHtIDsKM.........VGPAlKEVAAKnAGvkdA.................
PS5S       qdGEaLFKSKpCaACHsIDaKL.........VGPAFKEVAAKYAGQdGA.................
PS5M       asGEeLFKSKpCgACHsVQaKL.........VGPAlKDVAAKnAGvdGA.................
PS5D       stGEeLFKaKaCvACHsVDkKL.........VGPAFHDVAAKYgaQgdg.................
AV5        etGEeLYKTKgCtvCHaIDsKL.........VGPsFKEVtAKYAGQaGi.................

SG6     gDVaaGasVFSAN.CAACHmGGrNV.........IVan..KTLsKsd..LakYL.............kg
AA6     aDLahGgQVFSAN.CAsCHlGGrNV.........VnPa..KTLeKad..LDEY...............
AU6     iDInNGENIFTAN.CsACHaGGNNV.........IMPe..KTLkKda..LaDn...............
PR6     aDLdNGEkVFSAN.CAACHaGGNNa.........IMPd..KTLkKdv..LEan...............
BF6     aDIeNGErIFTAN.CAACHaGGNNV.........IMPe..KTLkKda..LEan...............
ML6     gDIaNGEQVFTtN.CAACHsvzZZk.........tLel..sSLwKaksyLaNF...............
EG6        gGaDVFadN.CstCHvnGgNV.........Isag..KvLsKta..IEEYL.............d.

PC54   aGdAaagEDkIgt..CvACHGtdgqG.........lApi....YPnLtGQsatYL...............
PS41   aGdAaagQakaav..CgACHGabbbG.........sApp....FPkLaGQgerYL...............
PS42   lfrggkiaEgMpa..CtgCHGsspvG.........iAta...gFPhLgGQhatYV...............
PSVM[2] agGgArsaDDiIakh.CnACHGagvlG.........apki..gdtaawkeradhqg.............gl

CF55    YDaAaGKatYDAs.CAmCHktGMMG.........APKv..GdkaaWapHIak.............GM
PH55[3] eqYDlAnGKtvYDAn.CAsCHaaGIMG.........APKt..GtarkWnsRLpq.............GL

DV53       adGaalYks..CigCHsadgg..........kammtnavkgkysdeelk................a
           1          10                   20          30
```

Figure 2 - Alignment of cytochromes c sequences in 39 species (see Fig. 1). The sequences are ordered and clustered as in (15). Too long sequences have been cut at their extremities : [1] gsd, [2] aas, [3] avtkadv, [4] aa, [5] pvaggea.

```
70          80          90                 100                 110         120
LtWTEaDLieYVtDPKpWLvkmTdDk.....gAKTKM..TFK...MgKNQa..DVvAFLaqnspdaggdge[4] PC50
LaWdEEhfvqYVqDPtkFLkeyTgDa.....KAKgKM..TFK...LkKEaDahNIwAYLqqVavrp       RF2S
faWTEEDIatYVkDPgaFLkekldDk.....KAKTgM..aFK...LaKggE..DVaAYLaSVvk         RF2C
LtWTEaNLaaYVknPKaFVlekSgDp.....KAKSKM..TFK...LtKDDEieNViAYLkTLk          QF2R
htWdDNaLkaYLlDPKgYVqakSgDp.....KAnSKM..iFR...LeKDDDvaNViAYLhTMk          QF2P
LvWTQENIiaYLpDPNaYLkkfltDkgqadkatgsTKM..TFK...LanDQQrkDVaAYLaTLk         RF2P

LTWDEaTfkeYItaPqkkV.............PGTKM..TFpG..LpNeaDrdNIwAYLsqfkaDGSK     RF2A
ITWtEevfreYIrDPKakI.............PGTKM..IFaG..IKDeQkVsDLIAYLKqfnaDGSKk    RF2V
IiWgEdTLmeYLeNPKkyI.............PGTKM..IFvG..IKkkeEraDLIAYLKkatne        HU
ITWgEdTLmeYLeNPKkyI.............PGTKM..IFaG..IKkksErvDLIAYLKdatsk        CH
IvWNEnTLmeYLeNPKkyI.............PGTKM..IFaG..IKkkgErqDLVAYLKSats         BN
LTWDEaTLdKYLeNPKavV.............PGTKM..VFvG..LKNPQDraDVIAYLKqlsgk        RD2
MTiDDamLtKYLaNPKetI.............PGnKMGAaFgG..LKNPaDVaaVIAYLKTvk          QF2M
MTiDEamLtNYLaNPKatI.............PGnKMGAsFgG..LKkPEDVkaVIeYLKTvk          QF2F
LTWDEpTLtKYLaDPKtvI.............PGnKM..VFaG..LKNPDDVkaVIeYLKTlk          QFM2
LTWDEpTLtKYLaDPKgvI.............PGnKM..VFaG..LKNPaDVaaVIAYLKSl           QFF2
IvWtpdvLfKYIehPqkiV.............PGTKM..gYpG..qpDPQkraDIIAYLeTlk          RF2G

eakLvaKVmaGgqGVWakqlg............aeIPM...PaN..nVTkEEAtrLvkWVLSlKqidyk     QF2T
palMAeRVRkGSvGIFG................kLPMtptPPa..rISDaDlKlViDWILktp          RFG2
eaeLAqRIKnGSqGVWG................pIPM...PPN..aVSDDEAqTLAkWVLSqK          PS5A
dktLAgHIKnGTqGnWG................pIPM...PPN..qVTDaEAlTLAQWVLSlK          PS5F
adlLAgHIKnGSqGVWG................pIPM...PPN..pVTEEEAKILAEWILSqK          PS5S
advLAgHIKnGStGVWG................aMPM...PPN..pVTEEEAKTLAEWVLTlK          PS5M
vahItnsIKtGSkGnWG................pIPM...PPN..aVSpEEAKTLAEWIVTlK          PS5D
adtLAaKIKaGgsGnWG................qIPM...PPN..pVSEaEAKTLAEWVLThK          AV5

fdddaVaAVaYQV..TN................GKNAM.PgFnG..RLSpkQIEDVAaYVVdQaEKGW      SG6
.gMaSIEAITTQV..TN................GKgAM.PAFGa..KLSaDDIEgVAsYaLdQSgKeW      AA6
.kMvSVNAITYQV..TN................GKNAM.PAFGS..RLaEtDIEDVANFVLTQSDKGWD     AU6
.sMnTIDAITYQV..qN................GKNAM.PAFGG..RLvDEDIEDaANYVLSQSEKGW      PR6
.gMnaVsAITYQV..TN................GKNAM.PAFGG..RLSDaDIEDVANYVLSQSEqGWD     BF6
..ngSesAIvYQV..TN................GKNAM.PAFGG..RLeDDEIaNVAsYVLSQag         ML6
.ggyTkEAIeYQV..rN................GKgpM.PAWeG..vLSEDEIvaVtDYVyTQaggaWanvs  EG6

....essIkayRDGqRkgg..............NaalMTpMaq...gLSDEDIAdIaaYySsqe         PC54
....lKqMndiKDGkRtvl..............ee..MTgLlt...bLSBZDIAaLadYaSqkmsvgmalbb[5] PS41
....aKqLtdfREGtRndd..............gtkiMqsIaai..kLSNkDIAaIssYiqglh         PS42
dgiLaKaIsgi.......................naM.ppkgtcadcSDDELreaiqkmSgl            PSVM

nvMVanSIkGY....KG................TKgmM.PAKGGNPkLTDaQVGNAVAYMVgQsk         CF55
atMIekSVaGYegeyRG................SKtfM.PAKGGNPdLTDkQVGDAVAYMVnEvl         PH55

ladymkaamgsakpvkgq...............gaeelykmkg....yadgsyggerkamskl           DV53
40          50                  60                  70          80
```

Figure 2 (continued) - When the homology is strong inside a family, the residue has been represented with a capital letter.

5 - if the new clustering is different from the old one, a new multiple alignment can be done following the new clustering (step 2). This process can be repeated until the clustering of the sequences is unchanged.

Technical description

The program is written in the Turbo Pascal language (Borland) and it runs on a MicroComputer with MS-DOS (Microsoft). Dynamic memory allocation is used throughout so that the number and size of sequences which can be handled is limited only by hardware and MS-DOS considerations.

Binary and Pascal codes for academic distribution are available from the author. A 5 1/4 or 3 1/2 inch diskette should be sent with request.

RESULTS AND DISCUSSION

The method has been used to align amino acid sequences of 39 related bacterial, algal and mitochondrial cytochromes *c*. The sequences have been extracted from the NBRF Protein Data Base (Release 15.0, January 1988). The gap penalty was set to 8 and the weight for substitution of an amino acid by another was given by Dayhoff's matrix (adding 8 to all entries). The weight for conservation of the methionine was increased from 14 to 18 because of its known importance in the holding of the heme. The initial pairwise comparisons have been executed by FASTP (17). With their scores, a first clustering and a first alignment has been done. Then the scores of the pairwise comparisons included in this alignment have been calculated and the new hierarchical clustering was as in Figure 1. A second iteration gave the alignment of Figure 2. The clustering was the same so no more iteration could be done. The alignment was produced entirely automatically, without any prealignment of key regions.

The results can be compared with a multiple alignment obtained by Dickerson (18) on evidence from X-ray structural analyses, so that structurally equivalent regions of the chain are aligned. Dickerson defined 7 families on criteria as length and origin of the bacteria : the hierarchical clustering of the sequences (Fig. 1) gives the same families as Dickerson's, except that P. mendocina c_5 is clustered with the cytochromes c_{555} and not with the cytochromes c_4.

Both alignments are similar. The residues that hold the heme are aligned in all the sequences (CxxCH near the beginning of the sequence and M near the end), except the methionine in the sequence of D. vulgaris c_{553} (Fig. 2). The massive deletions in the c_{551} are located at the folds of the c_2 proteins. Without the three-dimensional structures, the problems of where to place these deletions and of aligning the methionines seem to have been "insurmountable", to quote Dickerson (18), but the present program can do it.

The total score is not a good criterion for assessing the quality of a global multiple alignment as it is the sum of pairwise scores. The accuracy of an alignment of sequences has been defined by the percentage of residues that are aligned as in a reference alignment, obtained by crystallography, within test zones that have importance for the 3D structure (10). The accuracy of the alignment of Fig.2, calculated on 16 residues and with reference to Dickerson's alignment, is 95.1%, while the accuracy obtained with Barton & Sternberg's algorithm, is 92.8%.

With their algorithm or with Taylor's (9), the methionines that hold the heme, are aligned in every family but not between families. An advantage of the present algorithm over those that incorporate the sequences one by one in the final alignment, is that the sequences are aligned firstly inside the families; hence, when families are compared, the weight of already aligned residues is great and the best alignment is found when they are aligned. Feng and Doolittle utilize the same clustering scheme (13). In their method, the sub-cluster including the best scored pair of sequences, is not handled by the same algorithm as other clusters. This approach gives a major weight to the best scored pair, while other pairs of sequences could have nearly the same similarity score. Here, all clusters are aligned with the same algorithm.

It seems difficult to prove mathematically the convergence of the iterative process because the distance between two sequences, used to perform the clustering, is function of the global alignment. This convergence, however, has always been observed after one or two iterations, for all the treated examples.

The program can run on computers that use MS-Dos which are the most widespread in laboratories. On a 10 Mhz, 80-286 AT computer, it took 48 minutes to align the 39 sequences of cytochromes c, the length of which is around 150. The time required for an alignment by this algorithm is approximately proportional to $N(N-1)M^2$, where N is the number of sequences and M is the length of the sequences when aligned. This apply also to Barton and Sternberg's method, contrarily to what is written in their article (10).

CONCLUSION

The program described here allows one to find an alignment of many related sequences. It can be used either for proteins or for nucleic acids and it takes account of closer relationships that can exist among some subsets of sequences ; hence, it is attractive when there are subgroups in the family of sequences under study.

Global alignment of large numbers (50 to 250) of small sequences or smaller numbers of medium-length sequences (150 to 300) can be obtained easily.

ACKNOWLEDGEMENT

I thank Drs Daniel Kahn and Denis Corpet for their many helpful discussions.

REFERENCES

1. NEEDLEMAN, S.B. and WUNSCH, C.D. (1970) J. Mol. Biol. 48, 443-453.
2. MURATA, M., RICHARDSON, J.S. and SUSSMAN, J.L. (1985) Proc. Nat. Acad. Sci., U.S.A. 82, 3073-3077.
3. BAINS, W. (1986) Nucl. Acids Res. 14, 159-177.
4. SANKOFF, R.J. and CEDERGREN, G.L. (1976) J. Mol. Evol. 7, 133-149.
5. SOBEL, E. and MARTINEZ, H.M. (1986) Nucl. Acids Res. 14, 363-374.
6. MARTINEZ, H.M. (1988) Nucl. Acids Res. 16, 1683-1691.
7. SANTIBANEZ, M. and ROHDE, K. (1987) CABIOS 3, 111-114.
8. BACON, D.J. and ANDERSON, W.F. (1986) J. Mol. Biol. 191, 153-161.
9. TAYLOR, W.R. (1987) CABIOS 3, 81-88.
10. BARTON, G.J. and STERNBERG, M.J.E. (1987) J. Mol. Biol. 198, 327-337.
11. GRIBSKOV, M., MCLACHLAN, A.D. and EISENBERG, D. (1987) Proc. Natl. Acad. Sci. USA, 84, 4355-4358.
12. GRIBSKOV, M., HOMYAK, M., EDENFIELD, J. and EISENBERG, D. (1988) CABIOS 4, 61-66.
13. FENG, D-F. and DOOLITTLE, R.F. (1987) J. Mol. Evol., 25, 351-360.
14. DAYHOFF, M.O. (1978) In DAYHOFF, M.O. (ed), Atlas of Protein Sequence and Structure, National Biomedical Research Foundation, Washington, D.C., Vol 5, suppl. 3, pp 345-358.
15. BARTON, G.J. and STERNBERG, M.J.E. (1987) Protein Eng. 1, 89-94.
16. BENZECRI J.P. (ed) (1973) L'Analyse Des Données, Vol. 1, pp 153-206, Dunod, Paris.
17. LIPMAN, D.J. and PEARSON, W.R. (1985) Science 227, 1435-1441.
18. DICKERSON, R.E. (1980) Sci. Am. 242, 98-110.