ELSEVIER

Computers and Chemistry 26 (2002) 459–477

Computers & Chemistry

www.elsevier.com/locate/compchem

Local weighting schemes for protein multiple sequence alignment

Jaap Heringa [1]*

Division of Mathematical Biology, MRC National Institute for Medical Research (NIMR), The Ridgeway, Mill Hill, London NW7 1AA, UK

Received 29 May 2001; received in revised form 9 November 2001; accepted 20 November 2001

Abstract

This paper describes three weighting schemes for improving the accuracy of progressive multiple sequence alignment methods: (1) global profile pre-processing, to capture for each sequence information about other sequences in a profile before the actual multiple alignment takes place; (2) local pre-processing; which incorporates a new protocol to only use non-overlapping local sequence regions to construct the pre-processed profiles; and (3) local–global alignment, a weighting scheme based on the double dynamic programming (DDP) technique to softly bias global alignment to local sequence motifs. The first two schemes allow the compilation of residue-specific multiple alignment reliability indices, which can be used in an iterative fashion. The schemes have been implemented with associated iterative modes in the PRALINE multiple sequence alignment method, and have been evaluated using the BAliBASE benchmark alignment database. These tests indicate that PRALINE is a toolbox able to build alignments with very high quality. We found that local profile pre-processing raises the alignment quality by 5.5% compared to PRALINE alignments generated under default conditions. Iteration enhances the quality by a further percentage point. The implications of multiple alignment scoring functions and iteration in relation to alignment quality and benchmarking are discussed.

Keywords: Weighting schemes; Multiple sequence alignment; Profile

1. Introduction

The simultaneous alignment of three or more nucleotide or amino acid sequences is one of the most common tasks in bioinformatics. Multiple alignments are an essential pre-requisite to many further modes of analysis into protein families such as homology modelling, secondary structure prediction, phylogenetic reconstruction, or the delineation of conserved and variable sites within a family. Alignments may be further used to derive profiles (Gribskov et al., 1987) or hidden Markov models (Bucher et al., 1996; Eddy, 1998; Karplus et al., 1998) that can be used to scour databases for distantly related members of the family.

With the recent completion of the first draft of the human genome and genomes of many other species becoming rapidly available, the accurate alignment of biological sequences becomes more important than ever. Although many initiatives are underway for large-scale proteomics and structure elucidation of novel genomic proteins, an important drive in genomic approaches is essentially aimed at gathering the function of most if not all translated proteins from sequence data alone. To accomplish this, accurate alignment of the sequences is essential. Given the overwhelming amounts of sequence data, the alignment engines have to be extremely fast and fully automatic to be included in genomic pipelines.

Abbreviations: DP, dynamic programming; DDP, double dynamic programming; SP, sum-of-pairs.

* Tel.: +44-20-8959-3666x2293; fax: +44-20-8913-8545.
E-mail address: jhering@nimr.mrc.ac.uk (J. Heringa).

[1] www: http://mathbio.nimr.mrc.ac.uk

0097-8485/02/$ - see front matter
PII: S0097-8485(02)00008-6

The automatic generation of an accurate multiple alignment is potentially a daunting task. Ideally, one would make use of an in-depth knowledge of the evolutionary and structural relationships within the family but this information is often lacking or difficult to use. General empirical models of protein evolution (Benner et al., 1992; Dayhoff, 1978; Henikoff and Henikoff, 1992) are widely used instead but these can be difficult to apply when the sequences are less than 30% identical (Sander and Schneider, 1991). Further, mathematically sound methods for carrying out alignments, using these models, can be extremely demanding in computer resources for more than a handful of sequences (Carillo and Lipman, 1988; Wang and Jiang, 1994). To be able to cope with practical dataset sizes, heuristics have been developed that are used for all but the smallest data sets.

The most commonly used heuristic methods are based on the progressive alignment strategy (Hogeweg and Hesper, 1984; Feng and Doolittle, 1987; Taylor, 1988) with ClustalW (Thompson et al., 1994) being the most widely used implementation. The idea is to establish an initial order for joining the sequences, and to follow this order in gradually building up the alignment. Many implementations use an aproximation of a phylogenetic tree between the sequences as a guide tree that dictates the alignment order. Although, appropriate for many alignment problems, the progressive strategy suffers from its greediness. Errors made in the first alignments during the progressive protocol cannot be corrected later as the remaining sequences are added in.

Attempts to minimize such alignment errors have generally been targeted at global sequence weighting (Altschul et al., 1989; Thompson et al., 1994), where the contribution of individual sequences are weighted during the alignment process. However, such global sequence weighting schemes carry the risk of propagating rather than reducing error when used in progressive multiple alignment strategies (Heringa, 1999), for reasons given later in this paper.

The main alternative to progressive alignment is the simultaneous alignment of all the sequences. Two such implementations are available, MSA (Lipman et al., 1989) and DCA (Stoye et al., 1997). Both methods are based on the Carillo and Lipman algorithm (Carillo and Lipman, 1988) to limit computations to a small area in the multi-dimensional search matrix. They nonetheless remain an extremely CPU- and memory-intensive approach, applicable only to about nine sequences of average length for the fastest implementation (DCA). Iterative strategies (Hogeweg and Hesper, 1984; Gotoh, 1996; Notredame and Higgins, 1996; Heringa, 1999) are an alternative to optimise multiple alignments by reconsidering and correcting those made during preceding iterations. Although such iterative strategies do not provide any guarantees about finding optimal solutions, they are reasonably robust and much less sensitive to the number of sequences than their simultaneous counterparts.

All of these techniques perform *global* alignment and match sequences over their full lengths. Problems with this approach can arise when highly dissimilar sequences are compared. In such cases global alignment techniques might fail to recognise highly similar internal regions because these may be overshadowed by dissimilar regions and high gap penalties normally required to achieve proper global matching. Moreover, many biological sequences are modular and show shuffled domains (Heringa and Taylor, 1997), which can render a global alignment of two complete sequences meaningless. The occurrence of varying numbers of internal sequence repeats (Heringa, 1998) can also severely limit the applicability of global methods. In general, when there is a large difference in the lengths of two sequences to be compared, global alignment routines become unwarranted. To address these problems, Smith and Waterman (1981) early on developed a so-called *local* alignment technique in which the most similar regions in two sequences are selected and aligned. The algorithm has been extended in various techniques to compute a list of top-scoring pair-wise local alignments (Waterman and Eggert, 1987; Huang et al., 1990; Huang and Miller, 1991). Alignments produced by the latter techniques are non-intersecting; i.e., they have no matched pair of amino acids in common. For multiple sequences, the main automatic methods include the Gibbs sampler (Lawrence et al., 1993), MEME (Bailey and Elkan, 1994) and Dialign2 (Morgenstern, 1999). These programs often perform well when there is a clear block of ungapped alignment shared by all of the sequences. They perform poorly, however, on general sets of test cases when compared with global methods (Thompson et al., 1999a; Notredame et al., 2000).

Here, three strategies aimed at combining the best properties of global and local multiple alignment are presented: global pre-processing (Heringa, 1999) and two new strategies, local pre-processing and local–global DDP. The latter is a technique to integrate local alignment patterns into global alignment using the double dynamic programming (DDP) protocol. The three modes are incorporated in the multiple alignment package PRALINE (Heringa, 1999) and are used in progressive multiple alignment. They each incorporate a weighting scheme that is more flexible and appropriate for alignment than the aforementioned global sequence weighting schemes. The global and local pre-processing strategies are aimed at minimising error at early stages during progressive multiple alignment and do this by using information from other reliable sequences for each pair-wise comparison at every stage during progressive alignment. An important consequence of the

pre-processing scenarios is that they allow the calculation of a reliability index for each amino acid in the alignment, because the consistency of each aligned residue can be estimated based on the information from other sequences. The schemes allow further optimisation in iterative steps, based on the residue-specific reliability indices as an objective function. It will be shown how these weighting schemes can be used to refine multiple alignment, and how the iterative mode can further enhance the alignment quality. Also a new motif-based weighting scheme will be introduced here, based on a double dynamic programming to reconcile all best local alignments in a single global alignment. Finally, an evaluation of the weighting schemes with various parameter settings will be presented. Measuring the alignment quality is performed using the BAliBASE multiple alignment benchmark set (Thompson et al., 1999b), a database comprising five distinct alignment categories developed especially for testing the quality of multiple alignment methods.

2. Overview of existing sequence weighting schemes

There are two basic modes of sequence weighting that can be distinguished for multiple sequence alignment: global and local sequence weighting. The latter can be subdivided into position specific sequence weighting and motif weighting. This section will present an overview of previously developed techniques falling in the above three classes.

2.1. Global sequence weighting

Early on Altschul et al. (1989) and Vingron and Argos (1989) proposed global sequence weighting as a means to deal with the fact that sets of sequences normally are unequally represented. To address this problem, global sequence weighting involves the assignment of a weight for each sequence that is used in deriving any average value from the multiple alignment of a set of sequences. The Altschul et al. weighting scheme is integrated in the multiple sequence alignment method MSA (Lipman et al., 1989), where the sequence weights are derived from a rooted phylogenetic tree. Sequences closer to the root in the tree are assigned a larger weight than those at the periphery. In contrast, Vingron and Argos (1989) do not use a tree but derive the sequence weights by calculating for each sequence the average distance from all others. More distant sequences (outliers) according to this criterion receive a larger weight than relatively similar sequences.

Thompson et al. (1994) derived the sequence weights in a profile directly from the branch lengths of a phylogenetic tree constructed with the Neighbour-Joining technique (Saitou and Nei, 1987). They used these weights during progressive alignment in the construction of profiles representing pre-aligned sequence groups. Independently, Lüthy et al. (1994) modified the profile search technique by employing the Voronoi-based weighting (Sibbald and Argos, 1990), a technique that exploits a notion of the neighbourhood around each sequence in sequence space. Both techniques make use of the BLOSUM62 matrix, constructed from ungapped alignment blocks (Henikoff and Henikoff, 1992). In line with this derivation of the BLOSUM62 matrix, Thompson et al. (1994) exclude from analysis all alignment positions with a percentage of gaps higher than a certain specified threshold. Such regions would be expected to constitute loop regions in the associated protein structures showing less consistent amino acid conservation patterns.

Henikoff and Henikoff (1994) derived global sequence weights in a tree-less way by averaging the weights derived for each alignment column by a simple weighting scheme based on the principle of maximum entropy. For each alignment column, the contribution of each occurring amino acid is linearly down-weighted to make the overall contribution of each amino acid type equal. For example, if a column would have five valines and only one leucine, the contribution for each of the sequences having a valine at the considered alignment position would be 1/5 of that of the sequence comprising leucine.

In principle it is a good idea to perform global weighting of multiple alignments aimed at increasing the contribution of more distant sequences as they carry more information at each alignment position. However, when sequence weighting is used in progressive multiple alignment, the increased chance of mistakes when aligning distant sequences can well lead to error progression (Heringa, 1999). Vogt et al. (1995) compared local and global alignments of pair-wise sequences with a data bank of structure-based alignments (Pascarella and Argos, 1992) and included a set of over 30 substitution matrices with optimised gap-penalties. The best global alignments were achieved with the Gonnet residue exchange matrix (Gonnet et al., 1992), resulting in 15% incorrect residue matching when sequences with 30% residue identity were aligned. The error rate quickly increased to 45% incorrect matches at 20% residue identity of the aligned sequences, and to 73% error at 15% sequence identity. Rost (1999) stressed the same point and reported even higher pair-wise alignment error rates in the twilight zone. These statistics clearly demonstrate that increasing the global weight for distant sequences is likely to lead to the use of misalignment, which will hamper the recognition of true patterns. This is particularly significant if global weighting is used in progressive alignment, because incorrect alignments typically yield low scores, which make the involved sequences appear to be distant, such

that their incorrect contribution can be amplified by increased weighting in later progressive alignment steps.

2.2. Position-specific sequence weighting

Position specific sequence weighting involves the calculation of a weight for each alignment position. Although, many global weighting schemes derive their weights from averaging over positional weights calculated by position-specific schemes, the schemes are principally designed to differentiate the contribution of local alignment regions and aim to make use of the most informative fragments. Various schemes have been developed to locally adjust the contribution from the various sequences in an alignment, e.g. pseudocounts (Henikoff and Henikoff, 1994) or Dirichlet mixtures (Sjölander et al., 1996). These convert the observed amino acid frequencies into weights using background amino acid probabilities and residue exchange weights matrices (Pietroskovski, 1996).

Sunyaev et al. (1998, 1999) devised a strategy to construct profiles from given multiple alignments based on a weighting scenario reminiscent to phylogenetic parsimony methods. In their approach, amino acid propensities at each alignment position in the alignment profile are weighted according to the probability that identical amino acids occur in more than one sequence at the alignment position. If more alignment positions show identical conservation for a given subset of sequences (not necessarily the same conserved amino acid type over the alignment positions involved), the occurrence of the amino acids at those positions becomes more expected, which is corrected for by appropriately lowering the weight for the considered position. This approach leads to position-specific sequence weights, which are then implemented in the position-specific propensities for each of the amino acid types in a profile. The authors report increased sensitivity if profile searches were performed using profiles constructed with this technique. However, since their method is critically depending on which sequences are actually chosen to represent a given protein family in the multiple alignment, it is not suitable for progressive multiple alignment.

2.3. Motif-based weighting

An extension of local sequence weighting can be implemented in the $N \times M$ search matrix used in dynamic programming (DP), with N and M the length of the two sequences being compared. With DP, each cell in the matrix corresponds to a value representing the likelihood of the matched pair of amino acids from either sequence—typically taken from an amino acid exchange table—and can be weighted with any source of extraneous information.

Early on Argos (1987) weighted local diagonals in the alignment search matrix with correlations of physical–chemical amino acid characters such as hydrophobicity, bulkiness, size and the like. Each matched residue position in a local window received as a score a combination of the amino acid exchange value as given in the PAM-250 exchange matrix (Dayhoff, 1978), and the correlation of each of the five residue parameters calculated over a 5-residue window, each time with the considered matched pair at the middle window position. Although, Argos (1987) did not multiply align sequences but mainly used the scheme to generate visual dot plots for pair-wise protein sequence comparison, the approach basically weights local ungapped alignment scores with physical–chemical features.

More recently, Bucher and Hofmann (1996) developed a statistical local alignment technique for pair-wise sequence comparison in which each cell $[i, j]$ in the DP search matrix holds the total probability that a local alignment would pass through it. This is achieved by summing the scores of all local alignments intersecting cell $[i, j]$.

The multiple alignment method T-Coffee (Notredame et al., 2000) combines information from global and local pair-wise alignments. For each sequence pair, a single global alignment and 10 top-scoring non-intersecting local alignments are generated, respectively by the programs ClustalW (Thompson et al., 1994) and Lalign (Huang and Miller, 1991). The global and local alignment scores are then combined to yield a synthetic weight W for each aligned pair of amino acids, which is achieved by taking the sum of the associated basic scores (sequence identities): $W(A(x), B(y)) = \Sigma\, S(A(x), B(y))$, where $A(x)$ is residue x in sequence A, and summation is over the scores S of the global and local alignments containing the residue pair $(A(x), B(y))$, while for S each time the sequence identity percentage of the associated alignment is taken. This scenario results in a library of weights for each non-redundant residue pair. The information in the library is then further enhanced by a procedure called matrix extension (Notredame et al., 2000). Each library weight $W(A(x), B(y))$ is recalculated to reflect the degree to which residues $A(x)$ and $B(y)$ align consistently, as judged by all other library weights involving either $A(x)$ or $B(y)$. This is done using a triplet approach aimed at calculating the contribution of third sequences I onto the direct alignment of sequence A and B, based on the notion that a triplet alignment A–I–B effectively provides an alternative alignment of A and B. Each extended score W' is then calculated as $W'(A(x), B(y)) = W(A(x), B(y)) + \Sigma_{I \neq A,B}\, \mathrm{Min}(W(A(x), I(z)), W(I(z), B(y)))$, where x, y and z are sequence positions in sequences A, B and the intermediate sequence I, respectively, and summation is done over all third sequences I other than A or B. The

minimum of $W(A(x), I(z))$ and $W(I(z), B(y)$ is taken to use information from third sequences conservatively. The more intermediate sequences support the alignment of the pair, the higher becomes its extended weight. The extended library weights W' for each matched amino acid pair are then used to fill the DP search matrix and align the associated input sequences. Library extension is performed at each step during the progressive alignment, which is carried out following the ClustalW protocol (Thompson et al., 1994). The dramatic increase in sensitivity of the T-Coffee method is mainly a result of its matrix extension scenario, which combines local and global alignment, where an incorrect direct alignment of sequences A and B can effectively be overridden by consistent alignments of other sequences acting as intermediates in the above triplet alignments. Notredame et al. (2000) showed, using the BAliBASE set of reference alignments (Thompson et al., 1999a) as the standard of truth (see below), that T-Coffee generates much improved alignments as compared to ClustalW (Thompson et al., 1994), Prrp (Gotoh, 1996), and Dialign2 (Morgenstern, 1999): the overall relative improvements measured using the column score (see below) were 8.6, 8.6 and 17.2%, respectively.

3. Scoring alignments

Before the PRALINE alignment strategies and benchmarking results are described, it is convenient to first give an overview of existing modes of calculating similarity scores of pair-wise alignments, individual multiple alignments, and pair-wise multiple alignments, where the latter is normally used to benchmark multiple alignment routines with reference alignments.

3.1. Calculating pair-wise alignment scores

The alignment score of pair-wise sequence alignments is normally calculated, using a 20×20 amino acid exchange matrix and gap penalties, as the sum of the exchange values minus appropriate gap penalties:

$$S_{\mathrm{a,b}} = \sum_l s(a_i, b_j) - \sum_k N_k \cdot gp(k)$$

where the first summation is over the exchange values associated with l matched residues and the second over each group of gaps of length k, with N_k and $gp(k)$, respectively the number of gaps of length k and associated gap penalty. Using the common affine gap penalty scheme, a gap of length k is penalised with the value $gp(k) = \mathrm{pi} + k\ \mathrm{pe}$, where pi and pe are the penalties for gap initialisation and extension, respectively.

3.2. Sum-of-pairs score for single alignment

Since pair-wise alignment algorithms optimise the above score constituted by residue exchange values and gap penalties, an obvious way of scoring multiple alignments is to extend the pair-wise sequence scores to get a single score for a multiple alignment. This is referred to as the sum-of-pairs (SP) score for alignment. In the early simultaneous multiple alignment method MSA (Lipman et al., 1989), each cell in the multi-dimensional search matrix (see above), which corresponds to a column in the associated multiple alignment, is scored with the SP score. This requires special gap handling for matrix cells associated with gaps. Here, the SP score of a multiple alignment is calculated without additional penalties for gaps, consistent with the fact that gaps are also ignored in the sum-of-pairs score for pair-wise alignments (see below). For each amino acid $a_{i,x}$ in sequence i and at position x in the multiple alignment, the SP score is $\mathrm{SP}(x) = \Sigma_{k<l}\, s(a_{k,x}, a_{l,x})$, where $s(a_{k,x}, a_{l,x})$ is the amino acid exchange value. Alignment positions with gaps will be ignored in the pair-wise summation, which will effectively lower the resulting score. The overall SP alignment score is then calculated by summing over the alignment positions: $\mathrm{SP} = \Sigma_{1 \le x \le N}\, \mathrm{SP}(x)$, where N is the number of aligned positions.

3.3. Sum-of-pairs and column scores for comparing two alignments

In addition to the above SP score for single alignments, a pair-wise measure to determine the similarity between a query and reference alignment is also referred to as the sum-of-pairs score: $\mathrm{SP} = \Sigma_{1 \le x \le N}\, \mathrm{SP}(x)$, where N is the number of columns in the reference alignment and $\mathrm{SP}(x) = \Sigma_{k<l}\, \delta(r_{k,x}, r_{l,x})$, where $r_{k,x}$ is the amino acid in sequence k at position x in the reference alignment, and $\delta(r_{k,x}, r_{l,x}) = 1$ if the matched pair $(r_{k,x}, r_{l,x})$ is also matched in the query alignment; otherwise $\delta(r_{k,x}, r_{l,x}) = 0$. The score is commonly normalised using the total number of aligned amino acid pairs in the reference alignment. The SP score can also be weighted with amino acid exchange values and then normalised: $\mathrm{SP} = \Sigma_{1 \le x \le N}\, \Sigma_{k<l}\, \delta(r_{k,x}, r_{l,x})\, s(r_{k,x}, r_{l,x}) / \Sigma_{1 \le x \le N}\, \Sigma_{k<l}\, s(r_{k,x}, r_{l,x})$, where terms and indices are as above.

However, a more salient measure than the SP score for pair-wise alignment is the column score. In this measure, alignment columns of the query alignment are compared with those in the corresponding reference alignments and only taken as correctly reproduced if columns in query and target alignment are identical. The column score is given as the fraction of the reference alignment columns that is correct reproduced in the query alignment. Whereas, the SP score only gradually goes down with more misaligned sequences, a

single misaligned sequence can effectively zero the column score. Note that for the analysis presented below, the column score was used.

4. The PRALINE progressive alignment strategies

4.1. General outline

The PRALINE method (Heringa, 1999) relies on the Dynamic Programming (DP) technique for pair-wise sequence alignment, introduced by Needleman and Wunsch (1970) to the biological community. Input parameters for the DP algorithm are an amino acid substitution weights matrix and a gap opening and extension penalty value. The latter are applied each time when a gap is inserted in one of the sequences. Based on these parameters, the DP procedure is guaranteed to produce the optimal alignment of two sequences. For carrying out progressive multiple alignment with PRALINE, the basic DP routine is adapted for sequence–profile and profile–profile alignments. The PRALINE method does not use a pre-calculated search tree as do many progressive alignment methods (e.g. Hogeweg and Hesper, 1984; Thompson et al., 1994; Gotoh, 1996; Notredame et al., 2000) but performs at each alignment step a full profile search and compiles the optimal alignment score of the sequence block aligned in the preceding step with all other blocks and hitherto unaligned sequences. For the current alignment step it then selects the highest scoring pair of sequences or blocks of sequences to be aligned. The alignment order is thus established during the progressive alignment, such that a tree associated with the alignment order becomes only available upon completion of the alignment. The PRALINE method offers a number of strategies based on dynamic programming to optimise multiple alignment. Here, three of the strategies are included: global and local profile pre-processing, and local–global double dynamic programming. The first two strategies can be classified as position specific sequence weights, while the latter falls in the category of motif-based weighting schemes. Global and local profile pre-processing allow the calculation of a reliability index for each amino acid in the resulting multiple alignment, based on the consistency of pair-wise alignments. This, and iterative modes relying on the reliability indices, will also be described.

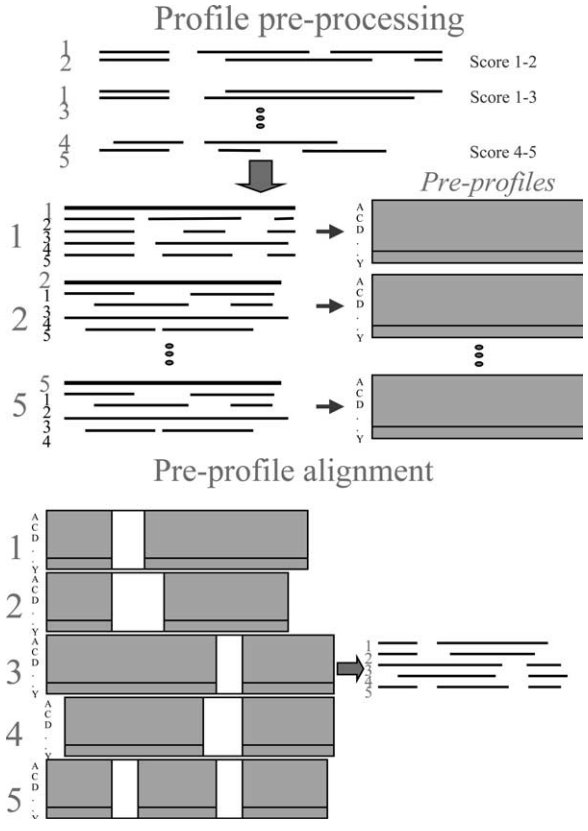

Fig. 1. Profile pre-processing and subsequent pre-profile alignment. The top panel shows a schematic outline of the construction of pre-processed profiles (pre-profiles) for five sequences. Pair-wise alignments are used to construct the five master-slaves alignments (pre-alignments), which are identified by the order number of their key sequence. The pre-profiles are compiled from the pre-alignments. The bottom panel shows how the pre-profiles are used to construct the final multiple alignment of the five original sequences. Note that for clarity, the pre-profiles depicted here all contain the total number of five sequences. In practice, setting the alignment score threshold value can lead to deletion of sequences from pre-processed blocks and associated pre-profiles.

4.2. Global profile pre-processing

The profile pre-processing strategy in the PRALINE method (Heringa, 1999) is a position-specific weighting scheme aimed at incorporating into each sequence, trusted information from other sequences. For each sequence, a multiple alignment is created by stacking other sequences (master-slaves alignment) that score beyond a user-specified threshold after pair-wise alignment with the sequence considered (Fig. 1). A low threshold would result in a pre-processed alignment for each sequence comprising all other sequences (where the chance for alignment error is large), while higher thresholds would allow the information from lesser sequences into the alignment (with fewer alignment errors). The use of a cut-off value for alignment scores, rather than alignment identity or length-normalised similarity values is in agreement with the analysis of Abagyan and Batalov (1997), who showed that alignment scores allow the best discrimination between alignments of structurally related and unrelated se-

Preprocessed profile for sequence 2:

```
2fcr            KIGIFFSTSTGNTTEVADFIGKTLGAKADAPIDVDDVTDPQALKDYDLLFLGAPTWNTGADTERSGTSWDEFLYDKLPEVDMKDLPVAIFGLGDAEGYPD
1fx1            KALIVYGSTTGNTEYTAETIARQL-ANAGYEVDSRDAASVEAFEGFDLVLLGCSTW--GDD---SIELQDDFLFDSLEETGAQGRKVACFGCGDS-SY-E
4fxn            -MKIVYWSGTGNTEKMAELIAKGISGKDVNTINVSDVNIDELLNE-DILILGC---SAMGDEVLEESEFEPFIEEISTKISGKKVALGSYGWGDGKWMRD
FLAV_ANASP      KIGLFYGTQTGKTESVaEIIRDEFGNDVVTLHDVSEVTD---LNDYQYLIIgCPTWNIG---ELQ-SDW-EGLYSELDDVDFNGKLVAYfGTGDQIGYAD
FLAV_AZOVI      KIGLFFGSNTGKTRKVaKSIKKRFDTMSDA-LNVNRVS-AEDFAQYQFLILgTPTLGPGLSSDCENESWEEFL-PKIEGLDFSGKTVALfGLGDQVGYPE
FLAV_CLOAB      KISILYSSKTGKTERVaKLIEE--GVKRSGNIEVKDAVDKKFLQESEGIIFgTPTYYANISWEMK--KW----IDESSEFNLEGKLGAAfSTANAGGSDI
FLAV_DESDE      KVLIVFGSSTGNTESIaQKLEELIAA-GGHEVTLLNAADASALADYDAVLFgCSAWGM-EDLEMQ----DDFLFEEFNRFGLAGRKVAAfASGDQE-Y-E
FLAV_DESGI      KALIVYGSTTGNTEGVaEAIAKTLNSEGTTVVNVADVTAPGLAEGYDVVLLgCSTW--GDDEIELQEDFVP-LYEDLDRAGLKDKKVGVfGCGDS-SY-T
FLAV_DESSA      KSLIVYGSTTGNTETAaEYVAEAFENK-EIDVELKNVTDVSVANGYDIVLFgCSTW--G---EEEIELQDDFLYDSLENADLKGKKVSVfGCGDSD-Y-T
FLAV_DESVH      KALIVYGSTTGNTEYTaETIAREL-ADAGYEVDSRDAASVEAFEGFDLVLLgCSTW--GDD---SIELQDDFLFDSLEETGAQGRKVACfGCGDS-SY-E
FLAV_ECOLI      AIGIFFGSDTGNTENIaKMIQKQLG--KDV-ADVHDISSKEDLEAYDILLLgIPTWYYG----EAQCDWDDF-FPTLEEIDFNGKLVALfGCGDQEDYAE
FLAV_ENTAG      TIGIFFGSDTGQTRKVaKLIHQKLDGIADAPLDVRRATREQFL-SYPVLLLgTPTLGDGLPGVEAGSSWQEFT-NTLSEADLTGKTVALfGLGDQLNYSK
FLAV_MEGEL      MVEIVYWSGTGNTEAMaNEIEAAVAAGADVSVRFED-TNVDDVASKDVILLgCPA--MGSE-ELEDSVVEPFFTDLAPK--LKGKKVGLfGYGWGSG---
3chy            KELKFLVVDDFSTRRIVRNLLKELGFNEEAEDGVDALNKLQA-GGYGFVI---SDWNM---PNMDGL---ELLKTIRADGAMSALPVLMV---TAEAKKE

2fcr            NFCDAIEEIHDCFAKQGAKPVGFSNPDDYDYEESKSVRDGKFLGLPLDMVNDQIPMEKRVAGWVEAVVSETGV
1fx1            YFCGAVDAIEEKLKNLGA----------------EIVQD----GLRID--GDPRAARDDIVGWAHDVRGAI--
4fxn            -FEERMNG-YGCVVVE--TPLIVQNEPD----EAE---------------QDCIEFGKKIANI----------
FLAV_ANASP      NFQDAIGILEEKISQRgGKTVGYWSTDGYDFNDSKALRNGKFVGLALDEDNQSDLTDDRIKSwVAQLKSEFGL
FLAV_AZOVI      NYLDALGELYSFFKDRgAKIVGSWSTDGYEFESSEAVVDGKFVGLALDLDNQSGKTDERVAAwLAQIAPEFGL
FLAV_CLOAB      ALLTILNHVKgMLVYSGG--VAFGKPKTHGYVHINEIQENE------D-ENARI-fGERiANkVKQIF-----
FLAV_DESDE      HFCGAVPAI-----EERAKELg-----------ATIIAEG--LKMEGDASND--P--EAVASfAEDVLKQL--
FLAV_DESGI      YFCGAVDVIEKKAEELgATLVA----------SSLKI-DGE-------------PDSAEVLDwAREVLARV--
FLAV_DESSA      YFCGAVDAIEEKLEKMgAVVIGDSLKIDGDPERDEIVSwGS--G-----IADKI-------------------
FLAV_DESVH      YFCGAVDAIEEKLKNLgA----------------EIVQD----GLRID--GDPRAARDDIVGWAHDVRGAI--
FLAV_ECOLI      YFCDALGTIRDIIEPRgATIVGHWPTAGYHFEASKGLADDHFVGLAID--EDRQPTAERVEKwVKQISEELHL
FLAV_ENTAG      NFVSAMRILYDLVIARgACVVGNWPREGYKFSFSAALENNEFVGLPLDQENQYDLTEERIDSwLEKL--KPAV
FLAV_MEGEL      EWMDAWKQRTE---DTgATVIG-----------TAIVNE-----MP-----DNAP-ECKElG--EAAAKA---
3chy            NIIAA--------AQAGAS--GY------------VVK--PFTAATLE--------EK-----LNKIFEKLGM

Iteration -1  SP= 127728.00  AvSP= 10.705  SId= 3764  AvSId= 0.315
```

Preprocessed profile for sequence 3:

```
4fxn            MKIVYWSGTGNTEKMAELIAKGIIESGKDVNTINVSDVNIDELLNEDILILGCSAMGDEVLEESEFEPFIEEISTKISGKKVALFGSYGWGDGKWMRDFE
1fx1            ALIVYGSTTGNTEYTAETIARQLANAGYEVDSRDAASVEAGGLFEGDLVLLGCSTWGDDSIEQDDFIPLFDSLETGAQGRKVACFGSYEYFCGA-VDAIE
2fcr            IGIFFSTSTGNTTEVADFIGKTL--GAKADAPIDVDDVTDPQALKDDLLFLGANTGADTERSGTSWDEFLYDKLPEVDMKDLPV-AIFGLGDAEGYPDFC
FLAV_ANASP      IGLFYGTQTGKTESVaEIIRD---EFGNDVVTLDVSQAEVTDLNDYQYLIIgCPTWNIGEL-QSDWEGLYSELDVDFNGKLVAYfGTIGYADNDAIGILE
FLAV_AZOVI      IGLFFGSNTGKTRKVaKSIKKRFDDETMS-DALNVNRVSAEDFAQYQFLILgTPTLGEGELENESWEEFLPKIGLDFSGKTVALfGQVGYPEGELYSFFK
FLAV_CLOAB      MKILYSSKTGKTERVaKLIEEGVKRSGNEVKTMNLDAVDKKFLQESEGIIFgTPTYYANI--SWEMKKWIDESSENLEGKLGAAfSTAGGSDIALLTILN
FLAV_DESDE      VLIVFGSSTGNTESIaQKLEELIAAGGHEVTLLNAADASAENLADYDAVLFgCSAWGMEDLEQDDFLSLFEEFNRGLAGRKVAAfAS---GDQEYVPAIE
FLAV_DESGI      ALIVYGSTTGNTEGVaEAIAKTLNSEGMETTVVNVADVTAPGLAGYDVVLLgCSTWGDDEIEQEDFVPLYEDLDAGLKDKKVGVfGSYTYFCGA-VDVIE
FLAV_DESSA      MSIVYGSTTGNTETAaEYVAEAFENKEIDVELKNVTDVSVADLGNYDIVLFgCSTWGEEEIEQDDFIPLYDSLNADLKGKKVSVfGDYTYFCGA-VDAIE
FLAV_DESVH      ALIVYGSTTGNTEYTaETIARELADAGYEVDSRDAASVEAGGLFEGDLVLLgCSTWGDDSIEQDDFIPLFDSLETGAQGRKVACfGSYEYFCGA-VDAIE
FLAV_ECOLI      TGIFFGSDTGNTENIaKMIQK---QLGKDVADVDIAKSSKEDLEAYDILLLgIPTYGEAQCDWDDFFPTLEEID--FNGKLVALfGDYAFCDAGTIRDIE
FLAV_ENTAG      IGIFFGSDTGQTRKVaKLIHQK-LDGIADA-PLDVRRATREQFLSYPVLLLgTPTLGDELVEASQYDSWQEFTNTDLTGKTVALfGNYSKNFVSAMRILY
FLAV_MEGEL      VEIVYWSGTGNTEAMaNEIEAAVKAAGADVESVRFEDTNVDDVASKDVILLgCPAMGSEELEDSVVEPFFTDLAPKLKGKKVGLfGSYGWGSGEWMDAWK
3chy            DKELKFLVVDDFSTMRRIVRNLLKELG--FNNVEEAEDGVD-ALNK-LQAGGYGVISDWNMPNMDGLELLKTI--RADGAMSALPVLMVTAEAKKENIIA

4fxn            ERMNGYGCVVVETPLIVQNEPDEAEQDCIEFGKKIANI
1fx1            EKLKNLGAEIVQDGLRIDGDPRAARDDIVGWAHDVRGA
2fcr            DAIEEHDCFAKQKPVGFSNPDDESKNDQIPMEKRVAGW
FLAV_ANASP      EKISGYGSKALRNGKFVGLALDEDNQDLTDDRIKVAQL
FLAV_AZOVI      DRTDGYEAVVVGLALDLDNQSGKTDERVAAwLAQIAPE
FLAV_CLOAB      HLMKgYGGVAFGKPYVHINEIQENEDENARfGERiANk
FLAV_DESDE      ERAKELgATIIAEGLKMEGDASNDPEAVASfAEDVLKQ
FLAV_DESGI      KKAEELgATLVASSLKIDGEPDSAE--VLDwAREVARV
FLAV_DESSA      EKLEKMgAVVIGDSLKIDGDPERDE--IVSwGSGIADI
FLAV_DESVH      EKLKNLgAEIVQDGLRIDGDPRAARDDIVGwAHDVRGA
FLAV_ECOLI      PRTAGYGLAFVGLAIDEDRQPELTAERVEKwVKQISEE
FLAV_ENTAG      DLVIARgCVVGNWPLLENNEPDQENQDLTELEKKPAVL
FLAV_MEGEL      QRTEDTgATVIGT-AIVNEMPDNA-PECKElGEAAAKA
3chy            AAQAGASGYVVK-PFTAATLEEKLNKIFEKLGM-----

Iteration -1  SP= 121196.00  AvSP= 10.075  SId= 3288  AvSId= 0.273
```

Fig. 2. Global profile pre-processing. An example is included of two pre-processed alignments for the sequences with PDB codes 2fcr and 4fxn. The sequences are taken from a data set of 13 flavodoxin sequences combined with the cheY sequence (PDB code 3chy). Pre-processing was effected with a score cut-off set at zero, thus allowing all remaining sequences in each pre-profile. Note that the key (or master) sequence in each of the two blocks (2fcr and 4fxn, respectively) contain no gaps, amino acids appearing opposite gaps in the key sequences are deleted from the added (slave) sequences in the pre-processed alignments.

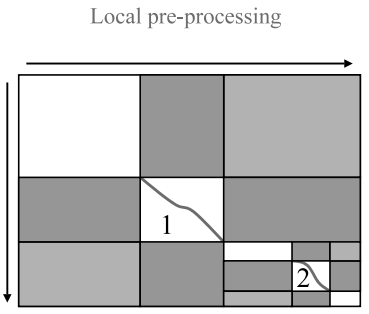

Fig. 3. Outline of local profile pre-processing. Shown are the first two highest-scoring local alignments with excluded associated matrix regions designated in grey: descriptions of the dark and light grey regions are given in the text. White matrix regions without alignments can either be filled with lower scoring top alignments, or remain blank if a threshold score value is used and no further local alignments score beyond the threshold.

quences. For each of the thus formed pre-processed alignments, a profile is constructed (Fig. 1). The PRALINE method then performs progressive multiple alignment using the pre-processed profiles, where each sequence is now represented by its pre-processed profile. To do this, the pair-wise alignment step is repeated over all pre-processed profiles, after which progressive alignment takes place. The pre-processed profiles for each of the sequences incorporate knowledge about other sequences (in particular similar sequences) and comprise position-specific gap penalties. This enables increased matching of distant sequences and likely placement of gaps outside the ungapped core regions in the pre-processed profiles during progressive alignment. While more details about global profile pre-processing can be found in Heringa (1999), Fig. 2 includes an example of two pre-processed sequence blocks. These are taken from a set of 13 distant flavodoxin sequences and the cheY sequence (Heringa, 1999), the latter being a signal transduction protein, which displays extremely low sequence similarity but nevertheless adopts the flavodoxin fold. The alignment score cut-off value can be specified by the user as the direct alignment score. Alternatively, it can be specified as a factor related to the aligned sequence lengths: $S \geq xL$, where x is the score threshold factor and L the length of the shortest matched sequence. This renders a score threshold that is linearly related to the alignment length, which is in agreement with the observation made for global similarity scores for random sequences (for a review, see Yona and Brenner, 2000).

4.3. Local profile pre-processing

The selection of sequences based on their overall pair-wise alignment scores can be extended by scrutinising local fragments within the sequences, and in addition to rejecting the information from low scoring sequences as can be done in global pre-processing, now also information can be discarded from sequence regions that cannot be trusted to contribute reliable information. This is achieved by a simple protocol that selects for each pre-processed profile the best sequence regions from sequences that are candidates for inclusion in the profile. The protocol is based on the local alignment algorithm of Smith and Waterman (1981). For each cell in the $N \times M$ search matrix, the following function is evaluated for two sequences $A = (a_1, a_2, \ldots, a_N)$ and $B = (b_1, b_2, \ldots, b_M)$:

$$H[i,j] = \mathrm{Max}\left\{\begin{array}{c} H[i-1,j-1] + s[a_i, b_j] \\ \mathrm{Max}\{H[i-x,j-1] - P(x)\} \\ \mathrm{Max}\{H[i-1,j-y] - P(y)\} \\ 0 \end{array}\right\} \qquad (2)$$

where $H[i,j]$ is guaranteed to hold the maximum similarity of two segments ending in a_i and b_j, $s[a_i, b_j]$ is the value for substitutions between residue types a_i and b_j, and $P(x)$ is the penalty for insertion of a gap of length x. Note that to convert a local alignment routine into a global algorithm, only the zero in (Eq. (2)) needs to be discarded. For local dynamic programming, the amino acids exchange matrix used must include negative values $s[a_i, b_j]$, otherwise global alignment will occur. The local alignment algorithm thus relies on dissimilar subsequences producing negative scores, which are subsequently discarded by placing zero values in the associated search matrix cells. After calculation of the search matrix by the above procedure, the local alignment corresponding to each non-zero cell in the search matrix can be obtained by a traceback procedure. Whereas, Smith and Waterman (1981) selected the optimal local alignment by performing the traceback only on the highest scoring matrix cell $H[i,j]$, Waterman and Eggert (1987) modified the traceback step to include a set of top-scoring non-intersecting alignments, i.e. having no matched amino acid pair in common. However, at each step in the traceback procedure for a local alignment that is being tested for inclusion in the top-scoring list, the currently matched residue pair must be checked against all matched pairs within the local alignments contained in the top-scoring list at that moment, which is computationally demanding. A result of this approach is that for a sequence A and B being compared, there can be more than one matched residue pair within the top-scoring alignments containing either residue $a_i (l \leq i \leq N)$ or $b_j (l \leq j \leq M)$. With our pre-processed profile approach with one line for each included sequence, this would lead to ambiguities in residue pair selection. Therefore, an alternative scenario was developed to avoid this selection problem and speed up computation at the same time. In this scenario, which is

outlined in Fig. 3, after performing the forward local dynamic programming step, the local alignment scores are ordered from high to low. Then the highest scoring alignment is selected and is traced back to obtain the pair-wise residue matches. The regions of the sequences corresponding to this alignment are then considered occupied by the local alignment and are effectively locked so to not allow any more alignments in this region (dark grey matrix regions in Fig. 3). Then the procedure is repeated for lower scoring local alignments that do not enter the locked matrix region. It is possible in principle to allow local alignments that would cross already accepted top alignments. For example, a local alignment j could include one sequence region positioned N-terminally of a sequence fragment within an accepted local alignment i, while the other sequence of j covers a region C-terminal of alignment i. However, since this would lead to motif inversions, which complicate the global multiple sequence alignment technique, these regions are disallowed by default in the local pre-processing procedure (light grey matrix regions in Fig. 3). This means that for each considered local alignment, the matched amino acid pairs (x, y) should be compared with all earlier accepted alignments. However, the comparison is more straightforward than in the approach by Waterman and Eggert (1987). If an earlier alignment would span residue a_i to a_j in sequence A and residue b_k to b_l in sequence B, either the condition ($x < a_i$ and $y < b_k$) should be met, or ($x > a_j$ and $y > b_l$). As a result of this local scenario carried out for each of the input sequences, the pre-processed profiles contain added sequences from which low scoring regions are deleted. The user must specify a local alignment cut-off score, so that only sequence fragments that locally align to the query sequence with a score above the cut-off value are included in the locally pre-processed profile. Fig. 4 shows an example of two locally pre-processed alignments, corresponding to those in Fig. 2 for global pre-processing. Due to the global relationship of the sequences, no sequence stretches matching middle regions of the key (or master) sequence have been discarded, but the use of information from the distant cheY sequence has been clearly reduced for the two sequences (Fig. 4).

4.4. Local–global DDP alignment

The local alignment-driven global alignment strategy, which falls in the class of motif-based weighting schemes (see above) operates in two steps (Fig. 5): First, for each possible residue match between two sequences (or sequence blocks), the score of the optimal local alignment including that match is calculated. Then, the optimal global alignment is compiled based on these local alignment scores. This two-step alignment strategy is akin to the double dynamic programming (DDP) protocol first implemented in the protein structure superpositioning algorithm SAP (Taylor and Orengo, 1989). The strategy can be viewed as a shortcut of the T-Coffee scenario described above (Notredame et al., 2000), as it ensures that the global alignment is biased towards matching local motifs, though using the local signals as soft constraints only. The strategy can be useful when local sequence similarity is suspected (for example in cases of very different sequence lengths). For each pair of sequences or sequence blocks, it uses the classical Smith and Waterman (1981) local alignment algorithm based on the dynamic programming protocol (Needleman and Wunsch, 1970). The idea is to determine for each matched pair of amino acids the score of the best local alignment over all those that include the pair. The thus obtained scores are then assigned to a search matrix as weights and subsequently subjected to a global alignment round, which resolves the values in a final global alignment that is biased towards local alignment.

In the Smith–Waterman algorithm, which is explained above, the local alignment corresponding to the highest scoring matrix cell $H[i, j]$ is determined by a traceback step. However, to meet our objective of calculating the score of the best local alignment for each matrix cell, for all or most of the $N \times M$ matrix cells, the traceback step would have to be performed and the corresponding score of the best local alignment substituted in the cell considered. This would be prohibitive in the context of multiple sequence alignment. Therefore, we devised the following shortcut to obtain the alignment score for each cell in the matrix without carrying out the traceback steps as shown in Fig. 5. Local dynamic programming is performed in forward and in backward direction of the sequences. Then, the values of the resulting two DP search matrices are added for each cell with subtraction of the local score $s[a_i, b_j]$ to avoid double counting of the local substitution value due to two times applying Eq. (2) for each cell $[i, j]$. After this operation, each cell in the resulting matrix shows the score of its best corresponding local alignment.

The thus obtained weights for each amino acid exchange of the two sequences are subjected to a second round of dynamic programming, this time to find the optimal global alignment (Gotoh, 1982) based on the local alignment scores (Fig. 5, step 2). A number of operations are available in the PRALINE method to scale the raw scores resulting from the local alignment step before the second global alignment step is effected. These include converting each score in the matrix to the z-score derived from the values in its corresponding row and column; converting the weights to logarithmic values; and adding the normalised weights to the residue exchange value corresponding to the search matrix cells.

Preprocessed profile for sequence 2: 2fcr

```
2fcr          KIGIFFSTSTSTGNTTEVADFIGKTLGAKADAPIDVDDVTDPQALKDYDLLFLGAPTWNTGADTERSGTSWDEFLYDKLPEVDMKDLPVAIFGLGDAEGYPD
1fx1          ...IVYGSTTGNTEYTAETIARQL---ANAGYEVDDAASVEAFEGFDLVLLGCSTW--GDDSELQ----DDFLFDSLEETGAQGRKVACFGCGDS-SY-E
4fxn          KI-VYWS-GTGNTEKMAELIAKGIGKDVNT-INVSDVNIDELLNE-DILILGCSA--MGDEVEES--EFEPF----IEEISTKGKKVALFGWGDGKGYG-
FLAV_ANASP    KIGLFYGTQTGKTESVaEIIRDEFGNDVVTLHDVSEVTD---LNDYQYLIIgCPTWNIG---ELQ-SDW-EGLYSELDDVDFNGKLVAYfGTGDQIGYAD
FLAV_AZOVI    KIGLFFGSNTGKTRKVaKSIKKTM---SDA-LNVNRVS-AEDFAQYQFLILgTPTLGEGSDCENE--SWEEFL-PKIEGLDFSGKTVALfGLGDQVGYPE
FLAV_CLOAB    KISILYSSKTGKTERVaKLIEE--GVKRSGNIEVKDAVDKKFLQESEGIIFgTPTY-------YANISWEKWI-DESSEFNLEGKLGAAfSTANSAGGSD
FLAV_DESDE    KVLIVFGSSTGNTESIaQKLEELIAAAADA--SAENLAD-----GYDAVLFgCSAWGM-EDLEMQ----DDFLFEEFNRFGLAGRKVAAfASGDQE-Y-E
FLAV_DESGI    ...IVYGSTTGNTEGVaEAIAKTLNSEGTTVVNVADVTAPGLAEGYDVVLLgCSTW--GDDIELQ----EDFLYEDLDRAGLKDKKVGVfGCGDS-SY-T
FLAV_DESSA    ...IVYGSTTGNTETAaEYVAEAFENK---EIDVENVTD-VSVADYDIVLFgCSTW--G---EEEIELQDDFLYDSLENADLKGKKVSVfGCGDSD-Y-T
FLAV_DESVH    ...IVYGSTTGNTEYTaETIAREL---ADAGYEVDDAASVEAFEGFDLVLLgCSTW--GDDSELQ----DDFLFDSLEETGAQGRKVACfGCGDS-SY-E
FLAV_ECOLI    ..GIFFGSDTGNTENIaKMIQKQLG-K-----DVADVHDKEDLEAYDILLLgIPTWYYG----EAQCDWDDF-FPTLEEIDFNGKLVALfGCGDQEDYAE
FLAV_ENTAG    .IGIFFGSDTGQTRKVaKLIHQKLDGIADAPLDVRRATREQFL-SYPVLLLgTPT--LG-DGELPGVSWQEFT-NTLSEADLTGKTVALfGLGDQLNYSK
FLAV_MEGEL    .VEIVYWSGTGNTEAMaNEIEKAAGADVESDTNVDDV----ASK--DVILLgCPA--MGSE-ELEDSVVEPFFTDLAPK--LKGKKVGLfGYGWGSG---
3chy          ..........................................................ADKELKFLVVDDFIVRNL----LKEL-----GFNNVEEAED

2fcr          NFCDAIEEIHDCFAKQGAKPVGFSNPDDYDYEESKSVRDGKFLGLPLDMVNDQIPMEKRVAGWVEAVVSETGV
1fx1          YFCDAIEE------K--LKNLG-----------AEIVQD----GLRID--GD--PRAARIVGWAHDV......
4fxn          --CVVVE-----------TPLIVQNPDE---AEQDCIEFGK..............................
FLAV_ANASP    NFQDAIGILEEKISQRgGKTVGYWSTDGYDFNDSKALRNGKFVGLALDEDNQSDLTDDRIKSwVAQLKSEFGL
FLAV_AZOVI    NYLDALGELYSFFKDRgAKIVGSWSTDGYEFESSEAVVDGKFVGLALDLDNQSGKTDERVAAwLAQIAPEFGL
FLAV_CLOAB    ---IALLTIH-LMVKSGG--VAFGKPKTHGYVHINEIQENE------D-ENARI-fGERiANkVKQI......
FLAV_DESDE    HFCGAVPAI-----EERAKELg-----------ATIIAEGKMEG---DASND--P--EAVASfAEDVLKQ...
FLAV_DESGI    YFCGAVDVIEKKAEELgATLVASSEPD------SAEVLD..............................
FLAV_DESSA    YFCGAVDAIEEKLEKMgAVVIGDSLKIDGDPERDEIVSwGS--G-----IADKI.................
FLAV_DESVH    YFCDAIEE------K--LKNLg-----------AEIVQD-----GLRID--GD--PRAARIVGwAHDV......
FLAV_ECOLI    YFCDALGTIRDIIEPRgATIVGHWPTAGYHFEASKGLADDHFVGLAID--EDRQPTAERVEKwVKQISEE...
FLAV_ENTAG    NFVSAMRILYDLVIARgACVVG--NPEGYKFSFSAALENNEFVGLPLDQENQYDLTEERIDSwLEAVL.....
FLAV_MEGEL    EWMDAWKQTED----TgATVIGTANPDN.............................................
3chy          G-VDALNKLQ-------AGGYGFSNMPNMDLELLKTIRDGAMSALPVLMVTAEAKKENIIAGYVAATLEE...
```

Preprocessed profile for sequence 3: 4fxn

```
4fxn          MKIVYWSGTGNTEKMAELIAKGIIESGKDVNTINVSDVNIDELLNEDILILGCSAMGDEVLEESEFEPFIEEISTKISGKKVALFGSYGWGDGKWMRDFE
1fx1          ..IVYGSTTGNTEYTAETIARQLANAGYEVDSRDAASVEAGGLFEGDLVLLGCSTWGDDSIEQDDFIPLFDSLETGAQGRKVACFGC---GDSSYVDAIE
2fcr          .KIIFFSSTGNTTEVADFIGKTL---GAKADAIDVDDVTDPQALKDDLLFLGAPTTGADT-ERSSWDEFLPEVDMK--DLPVAIF---GLGDAE------
FLAV_ANASP    ..LFYGTQTGKTESVaEIIRD---EFGNDVVTLDVSQAEVTDLNDYQYLIIgCPTIGE--L-QSDWEGLYSELDVDFNGKLVAYfGTIGYADGKWSTDFN
FLAV_AZOVI    ..LFFGSNTGKTRKVaKSIKKRFDETMSD--ALNVNRVSAEDFAQYQFLILgTPTLGEGELNESEFLPKIEGLD--FSGKTVALfGQVGYGEGSWSTD--
FLAV_CLOAB    MKILYSSKTGKTERVaKLIEEGVKRSGNEVKTMNLDAVD-KKFLQEEGIIFgTPTMKKWIDESSEFN--LEAfSTANSGSDIALLGGVAFGKPK------
FLAV_DESDE    ..IVFGSSTGNTEKLEELIAAG----GHEVTLLNAADASAENLADYDAVLFgCSAWGMEDLEQDDFLSLFEEFNRGLAGRKVAAfAS---GDQEY-EHFE
FLAV_DESGI    ..IVYGSTTGNTEGVaEAIAKTLNSEGMETTVVNVADVTAPGLAGYDVVLLgCSTWGDDEIEQEDFVPLYEDLDAGLKDKKVGVfGC---GDSSYTYDIE
FLAV_DESSA    ..IVYGSTTGNTETAaEYVAEAFENKEIDVELKNVTDVSVADLGNYDIVLFgCSTWGEEEIEQDDFIPLYDSLNADLKGKKVSVfGC---GDS----DYE
FLAV_DESVH    ..IVYGSTTGNTEYTaETIARELADAGYEVDSRDAASVEAGGLFEGDLVLLgCSTWGDDSIEQDDFIPLFDSLETGAQGRKVACfGC---GDSSYVDAIE
FLAV_ECOLI    ..IFFGSDTGNTENIaKMIQK---QLGKDV--ADVHDISKEDLEAYDILLLgIPTYGEAQCDWDDFFPTLEEID--FNGKLVALfGC---GD---QEDYA
FLAV_ENTAG    ..IFFGSDTGQTRKVaKLIHQGIADAPLDVRR-----ATREQFLSYPVLLLgTPTLGDELVEASQYDSWQEFTNTDLTGKTVALf---GLGDQNYSKNFV
FLAV_MEGEL    VEIVYWSGTGNTEAMaNEIEAAVKAAGADVESVRFEDTNVDDVASKDVILLgCPAMGSEELEDSVVEPFFTDLAPKLKGKKVGLfGSYGWGSGEWMDAWK
3chy          .RIV......N...LKEL---GFVEEAEDVDALNISDPNMDELLRADVLMVTAEAKKENIIAAAQVKPFLEEKLNKIFEK.....................

4fxn          ERMNGYGCVVVETPLIVQNEPDEAEQDCIEFGKKIANI
1fx1          EKLKNLGAEIVQDGLRIDGDPRAARDDIV.........
2fcr          ----GYPCDAIEKPVGFSN-PDDEESKSVRDGK.....
FLAV_ANASP    DSRNGVGLALDE-----DNQSDLTD-DRIEFG......
FLAV_AZOVI    ----GYEAVVVGLALDLDNQTDELAQIAPEFG......
FLAV_CLOAB    THL-GY----VHINEIQENEDENAR---I-fGERiAN.
FLAV_DESDE    ERAKELgATIIAEGLKMENDP-EAAEDVLK........
FLAV_DESGI    KKAEELgATLVASSLKIDGEPDSAE--VLDwAREVARV
FLAV_DESSA    EKLEKMgAVVIGDSLKIDGDPERDE--IVSwGSGIAD.
FLAV_DESVH    EKLKNLgAEIVQDGLRIDGDPRAARDDIV.........
FLAV_ECOLI    E----YFCDALGTDII---EP..................
FLAV_ENTAG    SAMRg-ACVVGNWPLLENNEPDQENQDLTE........
FLAV_MEGEL    QRTEDTgATVIGTAIV--NEPDNA-PECKElGE.....
3chy          ......................................
```

Fig. 4. Local profile pre-processing. An example is included of two pre-processed master-slaves alignments for the sequences with PDB codes 2fcr and 4fxn, corresponding to those in Fig. 2. The local alignment cut-off score was set at zero, so that all sequences are represented in each of the two pre-processed blocks and short local fragments are selected for some of the sequences. Deleted sequence regions are indicated by dots to discern them from gap regions within accepted local top alignments.

4.5. Consistency-based alignment iteration

The globally or locally pre-processed profiles can be used to derive consistency scores for each amino acid in the final multiple alignment, reflecting the consistency among the pair-wise alignments included in all the pre-processed profiles. The consistency scores are based on the extent to which the residue appears at the same alignment position across its corresponding identical sequences in the other pre-processed alignments (Fig. 1). The number of sequences allowed into the profiles during global pre-processing and the number of fragments selected during local pre-processing (Section 5) influences the consistency scores: the higher the cut-off values, the less corresponding identical amino acids there are within other pre-processed profiles, which in turn will tend to increase the consistency scores as fewer but more similar sequences or sequence fragments are being used. The consistency scores can be used in an iterative strategy, designed to give preference to consistent alignment regions in subsequent alignment rounds (Fig. 6, top panel: consistency iteration). This is achieved by using the reliability scores as residue weights in the alignments performed in the next iteration round (Heringa, 1999). Another possible iterative strategy updates the aligned sequences in the pre-processed alignments based on their placement in the multiple alignment of a preceding alignment round (Fig. 6, bottom panel: pre-profile update iteration). Although likely, iteration is not guaranteed to lead to convergence. The three possible outcomes of iteration are: (i) convergence to a single multiple alignment; (ii) limit cycle behaviour, where iteration after a number of iterations reaches an alignment that has been generated before, and then goes on repeating the iterative steps in between the two identical alignments; and (iii) divergence. The latter is declared when no conversion or limit cycle behaviour is reached. With the PRALINE method, the maximum number of iterations must be specified, but within this number of iterations, both convergence and limit-cycle behaviour is traced, upon which iteration is terminated. An additional option is to select, from all alignments constructed during iteration, the one with the highest SP score (see above). This option, called SP selection, is designed to control those iterations where alignments would wander away in alignment space to lower SP scores.

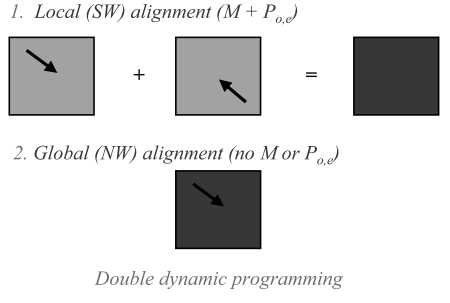

Fig. 5. Local–global DDP. (1) Summation of local DP search matrices resulting from forward and reversed local DP. For this step, a residue exchange matrix, and a gap open and extension penalty are required ($M + P_{o,e}$). (2) Second step of global DP to find the highest-scoring global alignment based on all best local alignment scores. No residue exchange matrix is needed for this step, and gap opening and extension penalty are set to zero (*no M or* $P_{o,e}$). For details, see text.

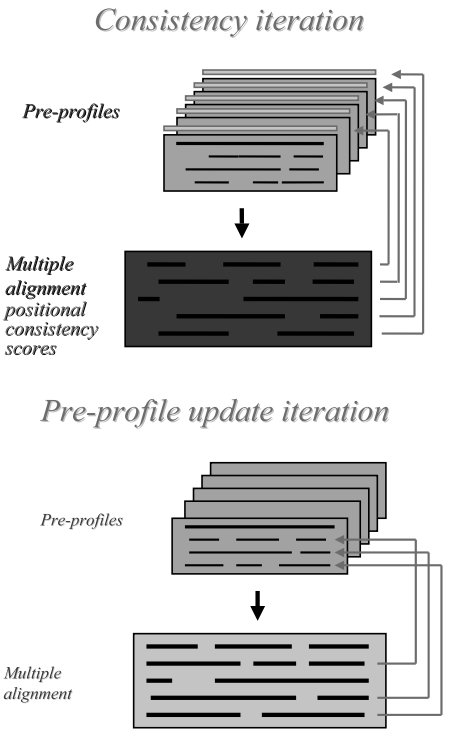

Fig. 6. Iteration of pre-profile multiple alignment. The top panel outlines schematically the consistency iteration protocol. The upward arrows at the right hand side indicate how the positional consistency scores for each multiply aligned sequence are copied into a vector for each pre-processed profile. The alignment can then be refined during subsequent iterations using the weights given in the profile vectors in each DP alignment during the progressive alignment protocol (downward arrow). After each alignment iteration, the consistency scores are recalculated and the profile weight vectors updated. The bottom panel depicts the pre-profile update iteration protocol. At each iteration, the pre-profiles are updated by using the multiple alignment resulting from the preceding iteration to position each sequence included in every pre-alignment under its key sequence. For each thus updated pre-alignment, a new pre-profile is constructed and a new round of progressive alignment is initiated.

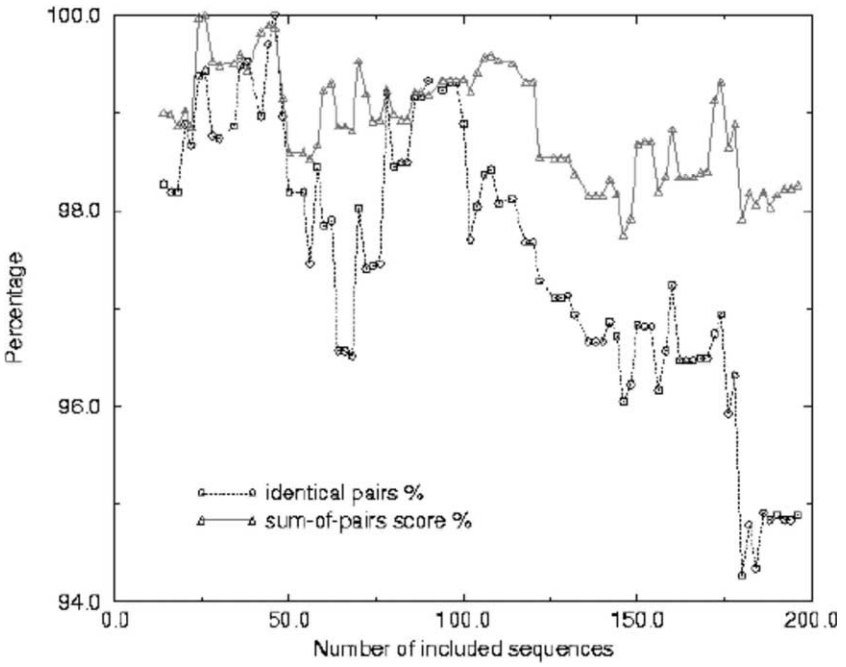

Fig. 7. Flavodoxin-cheY normalised multiple alignment Sum-of-Pairs (SP) scores versus total number of sequences included in the 14 pre-processed profiles resulting from varying the pair-wise alignment cut-off scores. The solid line (with triangles) denotes the SP scores, while the dashed line (squares) designates the number of identical pairs encountered in the alignments. Both the SP scores and the number of identical pairs scores have been normalised to the maximum found, multiplied by 100.

5. Tuning and evaluating the alignment strategies

5.1. Reference database

Evaluation tests were performed using the BAliBASE multiple alignment benchmark set (Thompson et al., 1999b) as the standard of truth. The BAliBASE alignments are placed in five different categories, which are supposed to cover most of the problems the alignment engines are faced with: (1) alignments containing equidistant sequences; (2) alignments with a single orphan sequence; (3) alignments comprising two distant groups; (4) alignments containing long deletions; and (5) alignments containing long insertions.

5.2. Tuning threshold values for profile pre-processed alignment

Can alignment score cut-off values be found that are generally optimal for global or local pre-processing? Fig. 7 shows, taking the flavodoxin–cheY data set as an example, the relation between the pair-wise alignment cut-off scores for global pre-processing and the SP scores (for single alignment) of each of the alignments produced. The alignments were made for a set of distant flavodoxin sequences and the extremely distant cheY sequence (Heringa, 1999), a signal transduction protein which nonetheless adopts the flavodoxin fold. The figure shows that the SP scores of the alignments do not display a uniform behaviour: A similar pattern is found for the number of identical pairs found in each of the alignments (Fig. 7). The variation of the SP scores is slightly over 2% and that of the identities about 6%. The flavodoxin-cheY alignment example illustrates that it is unlikely to derive an easy optimisation protocol for setting a fixed cut-off value for global pre-processing in individual alignments. However, Fig. 8 shows the pair-wise alignment scores of all possible sequence pairs in each of the BAliBASE alignments versus the length of the shortest sequence in each of the pairs, displaying a linear relationship. Consequently, global pre-processing was tested with alignment score threshold values specified linearly related to the sequence length: $S \geq xL$, where S is the alignment score, x the specified cut-off value and L the length of the shortest sequence. This means that the higher the value of x, the higher the alignment score S needs to be in order for the target sequence to be included in the pre-processed profile. The same holds for the threshold value for local profile pre-processing, although to a lesser extent, because here a more gradual effect between the threshold value and inclusion of sequence fragments is observed (data not shown). Moreover, local similarity scores are known not to grow linearly

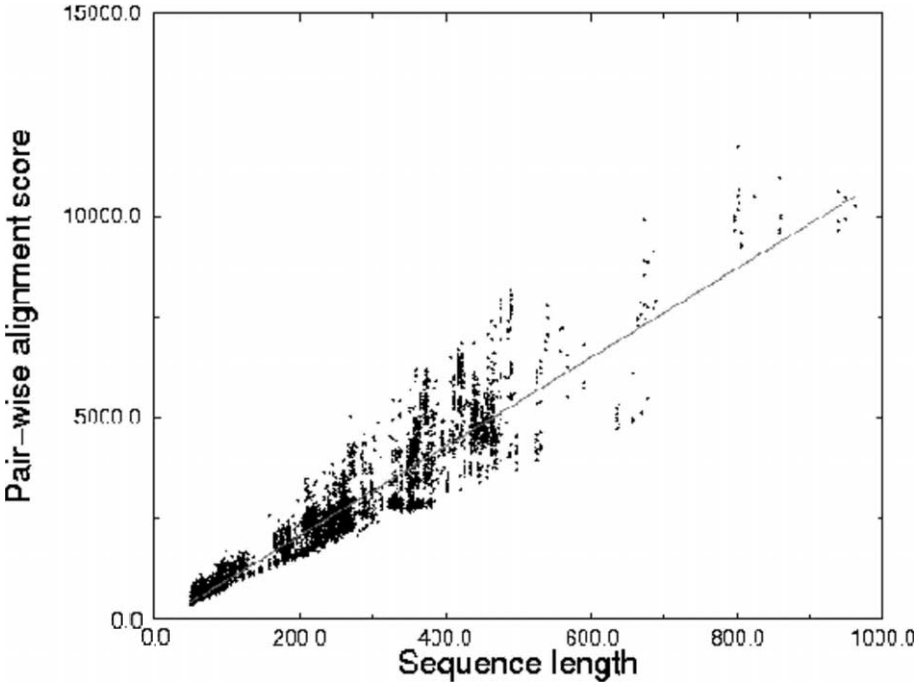

Fig. 8. BAliBASE pair-wise alignment scores versus the minimum sequence length of each sequence pair, aligned using the BLOSUM62 amino acid exchange matrix and penalties of 12 and 1 for gap opening and extension, respectively. The slope of the regression line is 11.0.

with the sequence length, but with the logarithm of the product of the sequence lengths (Karlin and Altschul, 1990).

5.3. Measuring alignment accuracy

Over a total of 144 BAliBASE reference alignments (version of July 2000), alignments generated by the PRALINE method were compared using column scoring (see above), i.e. alignment columns of the target alignments were compared with those in the corresponding reference alignments and were only taken as correct if columns were identical. Following Notredame et al. (2000), the column scores were evaluated only over alignment regions that were deemed to be reliable by the BAliBASE curators, who defined for each BALiBASE alignment the trusted regions. The PRALINE strategies were evaluated using a 500 MHz Pentium III cluster under the Linux operating system. We tested global and local profile pre-processing over the 144 BAliBASE alignments, and also ran the PRALINE consistency iteration protocol, with and without SP selection (see above).

5.4. Accuracy of global pre-processing

Table 1 shows the accuracy numbers generated under a number of score cut-off values for global pre-processing. It is clear that global profile pre-processing increases the quality of the alignments produced, compared to the PRALINE default setting without pre-processing, a maximum gain of 3.5% (weighted average) and 6.5% (unweighted average) for $S \geq L$ 9.5, where S is the alignment score and L the length of the shortest sequence in the pair-wise alignment. The greatest variation in accuracy percentages is observed for the BAliBASE categories 4 and 5 (long deletions and insertions, respectively).

The weighted average accuracy is further increased by 2.5% when iteration is applied. The best result is obtained using iteration and SP selection for $S \geq L$ 9, resulting in 65.35% weighted and 56.68 unweighted accuracy. SP selection enhances the accuracy in all categories except for $S \geq L$ 9.5, which shows a decline in accuracy of nearly 3% for BAliBASE category 4. Overall, the best scores for each of the categories was reached when iteration was performed, except for category 5, where both $S \geq 0$ and $S \geq L$ 9.5 attain the highest accuracy of 76.12%. Also, iteration for category 5 results in lower accuracy values for all tested cases. Although, SP selection improves the iteration by 6.5% or more, it does not reach the accuracy values observed for the non-iterative runs.

5.5. Accuracy of local pre-processing

Local pre-processing was tested by varying the alignment cut-off score of the local alignments, results of

Table 1
Benchmarking various Praline global pre-processing conditions

Method	Cat 1 (82)	Cat 2 (23)	Cat 3 (12)	Cat 4 (15)	Cat 5 (12)	Total 1 (144)	Total 2 (144)
default	77.07 ± 27.14	27.04 ± 19.95	49.38 ± 20.50	29.13 ± 36.17	59.53 ± 30.00	60.32 ± 34.31	48.43 ± 18.84
S≥0[a]	78.37 ± 28.27	28.80 ± 22.66	45.26 ± 21.53	18.93 ± 30.83	76.12 ± 23.22	61.31 ± 35.79	49.50 ± 24.18
S≥L*8	77.45 ± 30.03	28.05 ± 23.00	52.07 ± 23.31	27.47 ± 39.28	74.37 ± 27.69	61.98 ± 36.51	51.88 ± 21.56
S≥L*8.5	78.71 ± 27.12	32.54 ± 24.95	54.24 ± 26.29	27.67 ± 37.13	76.12 ± 23.22	63.76 ± 34.68	53.86 ± 21.23
S≥L*9	78.89 ± 26.68	32.03 ± 22.26	52.07 ± 18.79	31.21 ± 39.16	74.37 ± 27.69	63.83 ± 34.06	53.71 ± 20.20
S≥L*9.5	77.71 ± 26.83	31.27 ± 24.42	54.28 ± 17.64	37.88 ± 41.71	73.76 ± 25.48	63.86 ± 33.63	54.98 ± 18.57
S≥L*10	77.72 ± 27.55	29.61 ± 19.13	46.11 ± 23.07	35.80 ± 39.27	70.42 ± 27.77	62.43 ± 34.15	51.93 ± 18.97
S≥L*11	76.96 ± 27.18	27.52 ± 19.65	49.65 ± 17.56	27.33 ± 36.79	64.13 ± 25.28	60.55 ± 34.00	49.12 ± 19.71
S≥L*12	76.96 ± 27.06	26.84 ± 19.13	48.59 ± 23.63	27.47 ± 36.89	63.38 ± 28.31	60.30 ± 34.52	48.65 ± 19.71
S≥0 Wghtiter	79.10 ± 27.62	29.93 ± 22.49	38.27 ± 25.65	34.68 ± 37.34	68.89 ± 21.43	62.37 ± 34.83	50.17 ± 19.90
S≥0 WghtiterSP	78.60 ± 27.71	28.80 ± 22.66	45.07 ± 21.30	34.68 ± 37.34	75.88 ± 23.26	63.05 ± 34.60	52.61 ± 20.80
S≥L*9 Wghtiter	79.41 ± 26.74	31.84 ± 22.67	54.39 ± 19.41	42.12 ± 40.87	66.01 ± 28.37	64.73 ± 33.46	54.75 ± 16.85
S≥L*9 WghtiterSP	78.91 ± 26.69	32.64 ± 22.06	55.62 ± 18.35	42.12 ± 40.87	74.12 ± 27.71	65.35 ± 33.10	56.68 ± 17.83
S≥L*9.5 Wghtiter	77.92 ± 26.93	30.93 ± 24.09	55.92 ± 19.58	45.04 ± 43.26	67.28 ± 33.26	64.28 ± 33.26	55.44 ± 16.48
S≥L*9.5 WghtiterSP	77.92 ± 26.86	33.47 ± 23.17	56.31 ± 18.99	42.12 ± 40.87	73.52 ± 25.50	64.92 ± 32.80	56.67 ± 17.24

In the column 'Method', S≥L*x means that only sequences are included in the pre-processed blocks and profiles whenever their alignment score with the key sequence is higher than x times the length of the shortest sequence in the aligned pair (L); 'Wghtiter' indicates that consistency-based iteration is applied; and 'WghtiterSP' that SP selection is performed during iteration. The pair-wise alignment scores were calculated using the BLOSUM62 residue exchange matrix with 12 and 1 as gap opening and extension penalties, respectively. Accuracy numbers are given as average ± S.D. The numbers in brackets in the BAliBASE category headers denote the number of alignments in each category. Total 1 is the weighted average accuracy over 144 alignments, while Total 2 designates the unweighted accuracy, averaged over the five categories.

[a] S≥0 is equivalent to S≥L*7, i.e. all BAliBASE inter-alignment pair-wise sequence alignments calculated by PRALINE, score on average at least seven points per alignment position: note that the BLOSUM62 matrix used was made non-negative by adding eight to all exchange values.

Table 2
Comparing various Praline local pre-processing conditions

Method	Cat 1 (82)	Cat 2 (23)	Cat 3 (12)	Cat 4 (15)	Cat 5 (12)	Total 1 (144)	Total 2 (144)
LS≥0	78.59 ± 26.18	30.65 ± 22.34	53.15 ± 25.79	37.76 ± 39.47	83.84 ± 21.98	65.00 ± 33.77	56.78 ± 21.30
LS≥50	79.14 ± 25.03	30.60 ± 22.20	59.43 ± 20.91	37.67 ± 39.47	83.84 ± 21.98	65.81 ± 32.96	58.14 ± 21.36
LS≥100	78.88 ± 26.25	30.20 ± 24.63	49.04 ± 24.91	37.67 ± 39.47	83.84 ± 21.98	64.74 ± 34.28	55.93 ± 21.67
LS≥300	76.62 ± 28.56	25.52 ± 21.54	50.59 ± 25.38	26.21 ± 35.76	77.73 ± 28.58	61.13 ± 35.96	51.33 ± 22.95
LS≥0 Wghtiter	77.94 ± 27.62	26.29 ± 21.38	39.87 ± 21.47	45.05 ± 36.95	72.22 ± 26.70	62.61 ± 34.27	52.27 ± 19.68
LS≥0 WghtiterSP	78.74 ± 26.15	30.65 ± 22.34	53.15 ± 25.79	41.91 ± 38.61	83.84 ± 21.98	65.51 ± 33.35	57.66 ± 20.63
LS≥50 Wghtiter	77.96 ± 26.78	25.84 ± 21.53	36.47 ± 24.12	45.05 ± 36.95	72.22 ± 26.70	62.27 ± 34.34	51.51 ± 20.28
LS≥50 WghtiterSP	79.24 ± 25.08	30.60 ± 22.20	59.43 ± 20.91	41.91 ± 38.61	83.84 ± 21.98	66.31 ± 32.55	59.00 ± 20.62

The table format is identical to that of Table 1. Under Methods, LS≥x means that local sequence fragments are accepted in the pre-processed alignments only if their corresponding local alignment score (LS), from alignment with the key sequence, has a value of x or greater. The compilation of the local alignments was done using the BLOSUM62 matrix and gap penalties of value 12 and 1 for gap opening and extension, respectively.

Table 3
ClustalW, T-Coffee and best PRALINE performance over the BAliBASE benchmark set

Method	Cat 1 (82)	Cat 2 (23)	Cat 3 (12)	Cat 4 (15)	Cat 5 (12)	Total 1 (144)	Total 2 (144)
ClustalW	78.15 ± 24.33	32.66 ± 22.26	48.64 ± 20.08	41.66 ± 42.85	61.33 ± 29.74	63.22 ± 32.52	52.48 ± 15.88
T-Coffee	77.42 ± 28.99	34.61 ± 23.74	51.34 ± 27.31	53.49 ± 37.77	92.23 ± 9.64	67.15 ± 33.30	61.82 ± 20.44
S ≥ L*9 WghtiterSP	78.91 ± 26.69	32.64 ± 22.06	55.62 ± 18.35	42.12 ± 40.87	74.12 ± 27.71	65.35 ± 33.10	56.68 ± 17.83
LS ≥ 50 WghtiterSP	79.24 ± 25.08	30.60 ± 22.20	59.43 ± 20.91	41.91 ± 38.61	83.84 ± 21.98	66.31 ± 32.55	59.00 ± 20.62
Best global prepro	79.41 ± 26.74	33.47 ± 23.17	56.31 ± 18.99	45.04 ± 43.26	76.12 ± 23.22	66.29 ± 26.95	58.07 ± 17.66
Best local prepro	79.24 ± 25.08	30.65 ± 22.34	59.43 ± 20.91	45.05 ± 36.95	83.84 ± 21.98	66.65 ± 23.45	59.64 ± 19.04
Best overall	79.41 ± 26.74	33.47 ± 23.17	59.43 ± 20.91	45.05 ± 36.95	83.84 ± 21.98	67.20 ± 26.35	60.24 ± 19.35

The table format is identical to Table 1. All entries under 'Method' other than ClustalW or T-Coffee refer to PRALINE results. 'Best global prepro' lists the best PRALINE scores reached for each BAliBASE category under global pre-processing (Table 1), 'Best local prepro' lists those under local pre-processing (Table 2), while 'Best overall' shows the best category scores over the global and local strategies tested.

which are given in Table 2. Overall, the added advantage of discarding the contribution from non-matching sequence fragments enhances the accuracy of the alignments beyond the level attained by global pre-processing. The best result is reached for LS ≥ 50, which allows the contribution from all local alignments resulting from the selection protocol but the shortest fragments, with an alignment accuracy of 65.8% (weighted average) and 58.14% (unweighted average). This amounts to a quality increase of 5.5 and 9.7% compared to PRALINE default conditions, for weighted and unweighted mean values, respectively.

Iteration was performed for local alignment score threshold values of 0 and 50. It is clear that selecting the aligment with the best SP scores during iteration (SP selection), prevents the alignments from wandering away from the reference alignment. As with global pre-processing, SP selection does not perform well for category 4 aligments: while for four of five BAliBASE categories higher accuracy values are obtained, alignments of category 4 do better under iteration without the SP selection criterion. The best overall result is obtained with LS ≥ 50 using iteration with SP selection, resulting in 66.3 weighted and 59.9% unweighted accuracy. However, for individual categories and compared to non-iterative runs, iteration only optimises category 4 by seven percentage points, whereas in the other categories, iteration does not lead to an improvement.

5.6. ClustalW, T-Coffee and PRALINE performance

Table 3 gives the accuracy values for ClustalW and T-Coffee methods as well as the overall best results of PRALINE. While it is clear that T-Coffee is the most reliable method overall, with values of 67.1 and 61.8% for weighted and unweighted mean accuracy, respectively, PRALINE comes within 1% point of the weighted mean when local pre-processing is used (LS ≥ 50) in conjunction with iteration under SP selection. While T-Coffee manages to get the best scores of the three contenders for the categories 2, 4 and 5, PRALINE attains the best scores for category 1 and 3. If the best overall category scores by PRALINE are taken, PRALINE attains the same overall weighted average accuracy as T-Coffee. The largest differences between T-Coffee and PRALINE occur for the categories 4 and 5. Due to the T-Coffee motif-based weighting scheme and zero gap penalties for its dynamic programming run over the extended matrices (see above), it is relatively easy for T-Coffee to insert large insertions–deletions during progressive alignment.

6. Discussion

6.1. PRALINE protocols

In this paper, three protocols for multiple alignment have been described: global and local pre-processing, and double dynamic local–global programming. Pre-processing is aimed at using information from trusted sequences to optimise the representation of each of the query sequences in a profile, thereby also creating position-specific gap penalties for each sequence, before the actual progressive alignment takes place. The pre-processing protocol is used to calculate a consistency score for each residue in the multiple alignment. The consistency scores can be used to probe the multiple alignment, but they are applied here in an iterative strategy, and used to increase the weights of the associated

residues in the next iteration round (Fig. 6). This leads to alignment biased to consistent regions, where less consistent intervening regions are more likely to be re-aligned. The iterative strategy in PRALINE, based on global or local pre-processing, thus relies on the consistency of the multiple alignment as compared to optimal pair-wise alignment of the query sequence set.

6.2. Evaluation criteria and standard of truth

Evaluating multiple alignment programs is a complex issue. First of all, there is no general agreement as to what the standard of truth should be. For instance, should an alignment be evaluated using evolutionary, structural, or functional criteria? Although, in closely related familial sequences these criteria are expected to lead to the same alignment, in more distant cases they can result in very different answers. Moreover, benchmarks are usually carried out using a set of reference alignments, so that the evaluation becomes crucially dependent on the quality of such a reference alignment database. Furthermore, different ways have been proposed to quantify the agreement between a proposed and a reference alignment, such as weighted or unweighted sum-of-pairs scores, column score, etc. A multiple alignment in a sense can be viewed as a somewhat desperate attempt to obtain a unified picture of the relatedness of a set of sequences by averaging out matched residues that possibly cannot be consistently matched over the entire lengths of the sequences. This is because evolution, through mutations, insertions and deletions of sequence fragments, works on spatially and temporary de-coupled molecules, so that sequence alignment incompatibilities can well arise under divergent evolution, particularly with widely diverged sequences.

6.3. SP scores as objective function

As in most bioinformatics methods, multiple alignment techniques have two major components: the search function and the cost (or objective) function. In many algorithms, the cost function is considered the central problem. However, in this work, both issues are addressed in the iterative scheme (search function), which is based on alignment consistency and optionally on SP scores (objective functions) to optimise alignment quality. The objective function should approach the biological truth and require an in-depth knowledge of the evolutionary and structural–functional relationships within each individual family, which might well turn out to be impossible to generalise in a single scheme. To illustrate this point for the widely used SP scoring system (for single alignments), SP scores were calculated for each of the BAliBASE benchmark alignments. These scores were then compared to corresponding SP scores of the alignments calculated under PRALINE default conditions; i.e. generated without pre-processing or iteration, by calculating the difference between the reference SP score and that of the corresponding PRALINE alignment ($\Delta SP = SP_{Ref} - SP_{PRALINE}$). For just over a quarter (27%) of the BAliBASE alignments, the corresponding PRALINE alignments attain larger SP scores, while for the larger alignments ($N_{seq} L > 4500$) the fraction grows to 52% (Fig. 9). This might be referred to as the 'Charlie Chaplin' problem.[2] It is clear that trying to optimise the SP score for alignments that already score higher than their corresponding reference alignments is not likely to lead to convergence to the latter alignments.

6.4. Alignment optimisation by iteration

Notwithstanding the above, it has been shown here that consistency-based alignment iteration bound by SP scores (SP selection), can well optimise the quality of the resulting alignments. The alignments are driven through alignment space during iteration by a consistency-based protocol (see above). The consistency measure is different from the SP scores in that it is not based on highly scoring matched residue pairs, according to an amino acid exchange matrix, but assesses each amino acid by comparing the agreement between a multiple alignment and the associated optimal pair-wise alignments. If this agreement is high for certain alignment areas, these will be upweighted and have a higher chance to be aligned again in a next iteration round, such that other regions will effectively be realigned based on their consistency values. However, the accuracy of the generated alignments is measured here by column scoring over BAliBASE reference alignments, which does not incorporate any notion of consistency, so that successive iterative alignments could show uniformly decreasing SP scores (for single alignment). In such a case, SP selection would effectively undo the iteration by selecting the first alignment. We found that SP selection leads to better iterative alignments in the majority of BAliBASE categories, and in all cases optimises the overall accuracy scores compared to iteration without SP selection. Iterating globally pre-processed alignments with SP selection leads to a further gain in accuracy over consistency iteration without SP selection, of less than a percentage point (Table 1). After local pre-processing, iteration with SP selection results in an overall further increase of up to 4% (weighted mean) or 7% (unweighted) over consistency-based iteration alone (Table 2). However, iteration after local pre-processing leads to lower overall accuracy values than corresponding non-iterative runs, such that itera-

[2] At the peak of his fame, Charlie Chaplin allegedly entered a Charlie Chaplin contest in disguise and became second.

tions which wander away in alignment space need to be controlled by SP selection. Moreover, for most individual BAliBASE categories, local pre-processing without iteration leads to alignments with the same accuracy as the iterative modes. On the other hand, category 4, for which the least accurate alignments are produced in general, is generally about 3 percentage points better aligned using iteration without SP selection (Table 2). These results underline the complex relationship between consistency-based iterative optimisation, with or without SP selection, and reference alignment-based column scoring. A quarter of the BaliBASE alignments have SP scores that are already lower than PRALINE default alignments, generated without pre-processing or iteration (Fig. 9). On the other hand, disregarding 'neutral' alignments with ΔSP scores close to zero, about half the BAliBASE alignments show higher SP scores, so that SP selection would be expected to drive these alignments closer to their associated references.

6.5. Redundancy in BAliBASE

Although the BAliBASE alignment set (October, 2000) comprises 144 alignments, the number of different biological cases is in fact much lower, because many protein families have been re-used through the various BAliBASE categories. This effectively renders the data set critically small for reliable performance evaluation of multiple alignment programs. A particular risk arises when the database is used to tune the internal parameters of an alignment method, as over-fitting becomes highly likely. This would lead to a method that is highly effective when faced with most of the BAliBASE alignments, but would show a steep drop in accuracy when calculating an alignment unseen in the training phase. For example, the CLUSTAL alignment suite comprises a large number of carefully handcrafted heuristics driven by an extensive internal parameter set, which could render it prone to over-training. The CLUSTAL parameters have been tuned over BAliBASE since the creation of the latter by the CLUSTAL authors (Higgins, personal communication; Notredame et al., 2000) (Table 3).

6.6. Using PRALINE

PRALINE is a toolkit designed to allow the user to experiment with various strategies. If profile pre-processing is performed, the user can vary the consistency stringency by selecting the threshold for inclusion of global or local alignments. Compared to T-Coffee and ClustalW, which have established internal parameter settings, PRALINE is able to make the best individual alignments compared to the BAliBASE alignments, when given proper global or local pre-processing threshold parameters. However, for a user who wishes

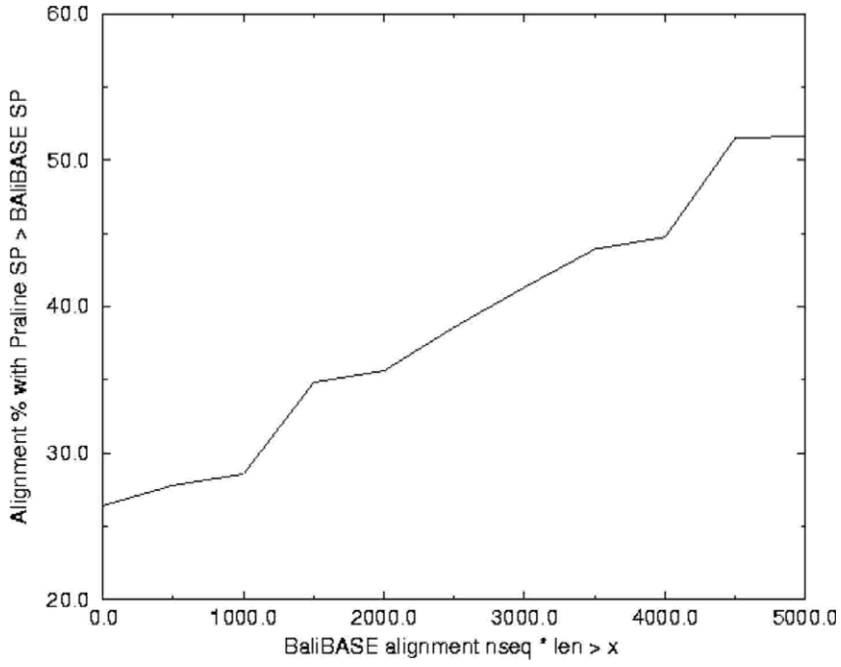

Fig. 9. Percentage BAliBASE alignments with $\Delta SP = SP_{Praline} - SP_{Ref} > 0$, i.e. for which the PRALINE alignments attain higher SP scores than their BAliBASE counterpart, versus the number of BAliBASE alignments with size $> x$. The size of the alignment is given as the product of the alignment length and the number of sequences.

to construct a multiple alignment without tuning, T-Coffee is currently the best general tool, as it is relatively fast, robust and generates alignments with a high and sustained quality (Table 3). It is clear that an interface, which would be able to recognise the type of alignment problem in the sense of the BAliBASE categories, and accordingly activate a PRALINE mode or the T-Coffee algorithm, would lead to further gains in accuracy.

6.7. Local–global DDP

Although, general benchmarking using the BAliBASE depository has yet to be carried out for the local–global DDP strategy, it has proven successful in a number of individual cases, for instance the alignment of the synaptobrevin family (Langosch, unpublished results). The approach is expected to do well in cases such as the categories 4 and 5 in the BAliBASE depository, which are the categories where T-Coffee currently shows significantly the best accuracy. Furthermore, the PRALINE method allows the combination of the local–global strategy with profile pre-processing, such that biasing alignments with local motifs can be complemented with information from the other sequences at each step of the progressive alignment.

6.8. Performance of PRALINE

The PRALINE approach is relatively slow when used with profile pre-processing. Whereas two full rounds of $N(N-1)/2$ pair-wise alignment are performed for pre-processing, and a total of $(N-2)(N-3)/2$ profile alignments during progressive alignment (with N the number of sequences), T-Coffee and ClustalW only require one round of pair-wise alignment for the guide tree and $N-2$ further alignments during building the multiple alignment. Although the iterative strategies in PRALINE enhance the alignment accuracy, they can slow down the method considerably depending on the number of iterations, which can be significant for large-scale alignment projects. Given the ability of the PRALINE method to compile alignments with very high quality, further work will be geared at a priori alignment characterisation aimed at taking the optimal strategy for the case at hand, and parallelisation of the code to speed up the pair-wise alignment stage. A webserver of the current implementation is available at http://mathbio.nimr.mrc.ac.uk

Acknowledgements

Nigel Douglas is thanked for expert handling of our computer resources, Cedric Notredame for helpful discussions, and Ruben Abagyan for his hospitality at the Scripps Resesarch Institute where part of this work has been carried out. Anonymous referees made helpful suggestions, which improved the manuscript.

References

Abagyan, R.A., Batalov, S., 1997. Do aligned sequences share the same fold? J. Mol. Biol. 273, 355–368.

Altschul, S.F., Carrol, R.J., Lipman, D.J., 1989. Weights for data related by a tree. J. Mol. Biol. 207, 647–653.

Argos, P., 1987. A sensitive procedure to compare amino acid sequences. J. Mol. Biol. 193, 385–396.

Bailey, T.L., Elkan, C., 1994. Fitting a mixture model by expectation maximisation to discover motifs in biopolymers. In: Proceedings of the second international conference on Intelligent Systems for Molecular Biology. AAAI Press, pp. 28–36.

Benner, S.A., Cohen, M.A., Gonnet, G.H., 1992. Response to Barton's letter: computer speed and sequence comparison. Science 257, 609–610.

Bucher, P., Karplus, K., Moeri, N., Hofmann, K., 1996. A flexible motif search technique based on generalized profiles. Comput. Chem. 20, 3–24.

Bucher, P., Hofmann, K., 1996. A sequence similarity approach based on a probabilistic interpretation of an alignment scoring system. In: States, D.J., Agarwal, P., Gaasterland, T., Hunter, L., Smith, R.F. (Eds.), Proceedings of the Fourth International Conference on Intelligent Systems for Molecular Biology (ISMB). AAAI Press, pp. 44–51.

Carillo, H., Lipman, D.J., 1988. The multiple sequence alignment problem in biology. SIAM J. Appl. Math. 48, 1073–1082.

Dayhoff, M.O. 1978. Atlas of Protein Structure and Sequence, National Biomedical Research Foundation, Washington DC, USA, 4, Suppl. 3.

Eddy, S.R., 1998. Profile hidden Markov models. Bioinformatics 14, 755–763.

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

Gotoh, O., 1982. An improved algorithm for matching biological sequences. J. Mol. Biol. 162, 705–708.

Gonnet, G.H., Cohen, M.A., Benner, S.A., 1992. Exhaustive matching of the entire protein sequence database. Science 256, 1443–1445.

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

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

Henikoff, S., Henikoff, J.G., 1992. Amino acid substitution matrices from protein blocks. Proc. Natl. Acad. Sci. USA 89, 10 915–10 919.

Henikoff, S., Henikoff, J.G., 1994. Position-based sequence weights. J. Mol. Biol. 243, 574–578.

Heringa, J., Taylor, W.R., 1997. Three-dimensional domain duplication, swapping and stealing. Curr. Opin. Struct. Biol. 7, 416–421.

Heringa, J., 1998. Detection of internal repeats: how common are they? Curr. Opin. Struct. Biol. 8, 338–345.

Heringa, J., 1999. Two strategies for sequence comparison: profile-preprocessed and secondary structure-induced multiple alignment. Comput. Chem. 23, 341–364.

eHogeweg, P., Hesper, B., 1984. The alignment of sets of sequences and the construction of phylogenetic trees. An integrated method. J. Mol. Evol. 20, 175–186.

Huang, X., Hardison, R.C., Miller, W., 1990. A space-efficient algorithm for local similarities. CABIOS 6, 373–381.

Huang, X., Miller, W., 1991. A time-efficient, linear-space local similarity algorithm. Adv. Appl. Math. 12, 337–357.

Karlin, S., Altschul, S.F., 1990. Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. Proc. Natl. Acad. Sci. USA 87, 2264–2268.

Karplus, K., Barrett, C., Hughey, R., 1998. Hidden markov models for detecting remote protein homologies. Bioinformatics 14, 846.

Lawrence, C.E., Altschul, S.F., Boguski, M.S., Liu, J.S., Neuwald, A.F., Wootton, J.C., 1993. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. Science 262, 208–214.

Lipman, D.J., Altschul, S.F., Kececioglu, J.D., 1989. A tool for multiple sequence alignment. Proc. Natl. Acad. Sci. USA 86, 4412–4415.

Lüthy, R., Xenarios, I., Bucher, P., 1994. Improving the sensitivity of the sequence profile method. Protein Sci. 3, 139–146.

Morgenstern, B., 1999. DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment. Bioinformatics 15, 211–218.

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

Notredame, C., Higgins, D.G., 1996. SAGA: sequence alignment by genetic algorithm. Nucleic Acids Res. 24, 1515–1524.

Notredame, C., Higgins, D.G., Heringa, J., 2000. T-Coffee: a novel method for fast and accurate multiple sequence alignment. J. Mol. Biol. 302, 205–217.

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

Pascarella, S., Argos, P., 1992. A data bank merging related protein structures and sequences. Protein Eng. 5, 121–137.

Pietroskovski, S., 1996. Searching databases of conserved sequence regions by aligning protein multiple-alignments. Nucleic Acids Res. 24, 3836–3845.

Rost, B., 1999. Twilight zone of protein sequence alignment. Protein Eng. 12, 85–94.

Saitou, N., Nei, M., 1987. The neighbor-joining method: a new method for reconstructing phylogenetic trees. Mol. Biol. Evol. 4, 406–425.

Sander, C., Schneider, R., 1991. Database of homology derived protein structures and the structural meaning of sequence alignment. Proteins 9, 56–68.

Sibbald, P., Argos, P., 1990. Weighting aligned protein or nucleic acid sequences to correct for unequal representation. J. Mol. Biol. 216, 819–836.

Sjölander, K., Karplus, K., Brown, M., Hughey, R., Krogh, A., Mian, I.S., Haussler, D., 1996. Dirichlet mixtures: a method for improved detection of weak but significant protein sequence homology. Comp. Appl. Biosci. 12, 327–345.

Smith, T.F., Waterman, M.S., 1981. Identification of common molecular subsequences. J. Mol. Biol. 147, 195–197.

Stoye, J., Moulton, V., Dress, A.W.M., 1997. DCA: an efficient implementation of the divide-and-conquer approach to simultaneous multiple sequence alignment. Comput. Appl. Biosci. 13, 625–626.

Sunyaev, S.R., Rodchenkov, I.V., Eisenhaber, F., Kuznetsov, E.N. 1998. Analysis of the position dependent amino acid probabilities and its application to the search for remote homologues. In: Proceedings of the 2nd Annual International Conference on Computers in Molecular Biology (RECOMB98), pp. 258–264.

Sunyaev, S.R., Eisenhaber, F., Rodchenkov, I.V., Eisenhaber, B., Tumanyan, V.G., Kuznetsov, E.N., 1999. PSIC: profile extraction from sequence alignments with position-specific counts of independent observations. Protein Eng. 12, 387–394.

Taylor, W.R., 1988. A flexible method to align large numbers of biological sequences. J. Mol. Evol. 28, 161–169.

Thompson, J.D., Higgins, D.G., Gibson, T.J., 1994. CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, positions-specific gap penalties and weight matrix choice. Nucleic Acids Res. 22, 4673–4680.

Thompson, J.D., Plewniak, F., Poch, O., 1999a. A comprehensive comparison of multiple sequence alignment programs. Nucleic Acids Res. 27, 2682–2690.

Thompson, J.D., Plewniak, F., Poch, O., 1999b. BAliBASE: a benchmark alignment database for the evaluation of multiple sequence alignment programs. Bioinformatics 15, 87–88.

Vingron, M., Argos, P., 1989. A fast and sensitive multiple sequence alignment program. Comp. Appl. Biosci. 5, 115–121.

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

Wang, L., Jiang, T., 1994. On the complexity of multiple sequence alignment. J. Comput. Biol. 1, 337–348.

Waterman, M.S., Eggert, M., 1987. A new algorithm for best subsequences alignment with applications to the tRNA-rRNA comparisons. J. Mol. Biol. 197, 723–728.

Yona, G., Brenner, S.E., 2000. Comparison of protein sequences and practical database searching. In: Higgins, D., Taylor, W.R. (Eds.), Bioinformatics: Sequences, structure and databanks, Practical Approach Series. Oxford University Press.