Multiple Alignment of Protein Structures in Three Dimensions

Evgeny Krissinel and Kim Henrick

European Bioinformatics Institute, Hinxton, Cambridge CB10 1SD UK,
keb@ebi.ac.uk,
WWW: http://www.ebi.ac.uk/msd-srv/ssm

Abstract. The paper describes the algorithm of multiple alignment of protein structures in 3D used in the EBI-MSD web service SSM (Secondary Structure Matching) located at URL given in the title. Structure alignment is known as a computationally hard procedure, with multiple alignment being considerably harder then a more conventional pairwise alignment. We base our approach on an efficient SSM algorithm for pairwise structure alignment, which allowed for multiple alignment of a considerably larger number of structures (up to 100), on comparison with alternative techniques, in real time.

1 Introduction

Comparison studies play an important role in structural biology. It is widely acknowledged that structural similarity is a clue for the identification of protein function and evolution. Often structural similarity is estimated by sequence identity, obtained in the course of sequence alignment, assuming that higher sequence similarity is a necessary condition for structures to be geometrically similar. Vast data on protein structures, accumulated in PDB over last decades, allow nowadays for a detail structure analysis. It was found (cf. Refs. [1–4]) that structural similarity is not a simple function of sequence identity. As appears, only 20% of identical residues in two chains is often sufficient for structures to be very similar.

This result implies that structure-related studies should use geometry-based tools whenever possible. A number of methods for the comparison of protein structures have been developed over last decade. Most of the effort was invested into algorithms for pairwise structure alignment [1, 5–18], but only a few techniques for the alignment of multiple structures in 3D have been reported [19–21].

In this paper, we describe the algorithm of multiple structure alignment employed in the EBI-MSD web-server SSM (found at URL given in the title). The server delivers both pairwise and multiple alignments of protein structures in 3D. The SSM's pairwise alignment algorithm was detailed in Ref. [1].

2 General notes

Multiple structure alignment (MA) may be defined as identification of residues that occupy geometrically equivalent positions in all (more than 2) aligned structures. Geometrically equivalent residues are found in close proximity of each other, when structures are properly rotated and translated (superposed). Evidently, there are many different rotations and translations that put some of residues into superposition, so there are many different alignments. From that manifold, we focus on alignments which maximise a certain score function, which normally depends on the number of superposed residues and a measure of distance between the superposed structures.

Although the above definition is identical to that of pairwise alignment (PA), it is important to realize that, in general, multiple alignment *does not* reduce to the set of all-to-all pairwise alignments of given structures. Identification of geometrical equivalence is always a subject to certain criteria, and unless structural similarity is high, a small distortion of one structure may noticeably change the pairwise alignment. As a result, if ith residue of structure A, r^i_A, may be aligned to residue r^j_B of structure B, and the latter – to residue r^k_C of structure C, it does not necessarily mean that residues r^i_A and r^k_C may be also aligned. Only in the simplest case of highly similar structures, when geometrical equivalence of residues is established well within the used geometrical criteria, multiple alignment is given by the intersection of all-to-all pairwise alignments.

It follows from the above that multiple alignment almost always results in lower pairwise scores, so that structures appear more distant then would be concluded from the pairwise comparisons. On the other hand, MA is biased to spotting out structural features that are common for all aligned structures, therefore one may expect that multiple alignments are less affected by artefacts of employed geometrical criteria.

The problem of multiple structure alignment, just as that of PA, does not have an exact solution, and any solution is subject to accepted definitions of structural similarities and scores. There are no common agreements on the latter. All known methods (cf. Refs. [19–21]) use different techniques that try to improve a starting alignment chosen from initial pairwise all-to-all alignments, typically by chosing a pair of most similar structures and consecutive addition of closest structural neighbours to the alignment. In Ref. [21], MA is sought from initial PAs by Monte-Carlo moves representing indels. Neither of techniques guarantees convergence to optimal solution. We suggest an approach, which is based on iterative removal of structural elements that have least chances to get aligned, according to a heuristic score. After all non-aligneable structural elements are identified, the solution is refined by iterative multiple alignment of backbone C_α atoms.

3 Algorithm

3.1 Multiple alignment of structural elements

In the following discussion, we define structural element as one or more secondary structure elements (SSE), found in a certain geometrical orientation to each other and ordered in the same way along aminoacid chain. We also assume that

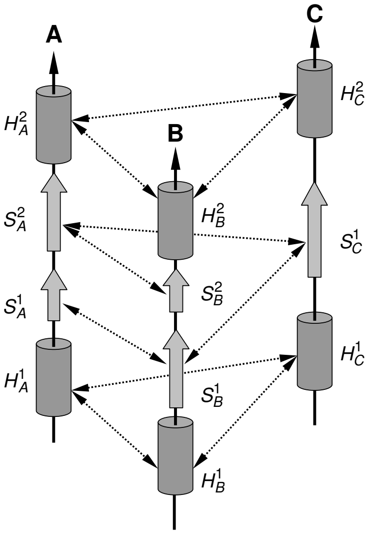

Fig. 1. Schematic of multiple alignment of structural elements ($H^k_{A,B,C}$ stand for helices and $S^k_{A,B,C}$ - for strands of chains A, B and C, respectively; arrows denote pairwise alignments of SSEs). Chains A and B may be unambiguously aligned, while there is an ambiguity of their alignment to chain C: any of strands $S^k_{A,B}$ may be aligned to S^1_C. Given that geometrically best pairwise alignments map S^2_A and S^1_B onto S^1_C, as shown in the Figure, the strands cannot be multiply aligned by a simple intersection of their pairwise alignments.

multiple alignment preserves the connectivity of structural elements, i.e. for any aligned pairs (S^i_A, S^j_B) and (S^m_A, S^n_B) (where superscripts denote the element's serial numbers and subscripts - chains) $\mathrm{sign}(m-i) = \mathrm{sign}(n-j)$.

The problem of multiple alignment of structural elements is illustrated in Fig. 3.1. In this illustration, multiple alignment of helices may be unambiguously obtained as an intersection of their pairwise alignments, while strands do not seem to align because strands S^2_A and S^1_B in structures A and B, aligned to the only strand S^1_C in structure C, do not align to each other.

The above consideration, however, does not mean that one should not *try* to align strands in this example. Indeed, if pairs (S^2_A, S^2_B) and (S^2_A, S^1_C) were found as geometrically equivalent, one can assume that pair (S^2_B, S^1_C) could be also aligned, however with a lower pairwise score than that of pair (S^1_B, S^1_C). Similar reasonings lead to the conclusion that pair (S^1_A, S^1_C) could be aligned as well. It is not a rare situation in protein structure comparison that a particular structure element of one structure may be equivalenced with more than one element of another structure; should that be the case, solution with maximal pairwise score is chosen [1].

Therefore, it may be suggested, that, having the results of pairwise all-to-all alignments as a starting point, one possibly needs to remap those structural elements that may be connected by PA relations (dotted arrows in Fig. 3.1), but do not multiply align as an intersection of PAs. For the schematic in Fig. 3.1, one would need to choose from 4 remappings: (S^i_A, S^j_B, S^1_C), $i, j = 1, 2$, the one which maximises a defined MA score. Being apparently correct in general, this simple recipe has two main drawbacks. Firstly, remapping of structural elements changes the optimal orientation of structures and, as a result, pairwise scores for all structural elements also change. For example, remapping of strands in Fig. 3.1 might make some helices non-matching. This means that structural elements should be remapped gradually, one-by-one, with recalculation of all pairwise alignments after each remapping. Secondly, the number of possible remappings depends exponentially on the number of aligned structures. Extension of example in Fig. 3.1 onto 11 chains gives 2^{10} possible remappings if each chain, except one, has only two candidate strands for alignment. In practice, this makes multiple alignment of more than 10-15 structures computationally prohibitive. In this situation, our suggestion is to gradually exclude structural elements, which have least chances to get aligned, from consideration. The chances are estimated by a heuristic score, as described below.

Algorithm of multiple SSE alignment

1. Initialise an empty list $\mathcal{L}$ of excluded SSEs.
2. Calculate $N(N-1)/2$ pairwise alignments between all N given structures.
3. For each SSE $\notin \mathcal{L}$, calculate the total number of SSEs in other structures it is aligned to, P_x^i (i stands for the SSE serial number in structure x). If $P_x^i = N-1$ for all i and x, then all SSEs not found in list $\mathcal{L}$ have been multiply aligned and algorithm quits. Otherwise, proceed to step 4.
4. For each SSE $\notin \mathcal{L}$ with $P_x^i < N-1$, calculate the alignment score Q_x^i. We define this score as a sum of Q-scores in all pairwise alignments for the given SSE (cf. Eqs. (8,10) in Ref. [1]):

$$Q_x^i = \sum_y \sum_j \frac{\left(N_{xy}^{ij}\right)^2}{\left(1+\left(RMSD_{xy}^{ij}/R_0\right)^2\right) N_x^i N_y^j} \tag{1}$$

   where y enumerates structures, j enumerates SSEs in a structure, N_{xy}^{ij} is the number of aligned residues in ith SSE of structure x and jth SSE of structure y, $RMSD_{xy}^{ij}$ - r.m.s.d. of aligned residues, N_x^i and N_y^j are the total numbers of residues in the SSEs. R_0 is an empirical parameter measuring the importance of r.m.s.d. versus the alignment length, chosen at 3 Å [1].
5. Identify the least Q_x^i and place ith SSE of structure x into list $\mathcal{L}$. If all SSEs of structure x are found in list $\mathcal{L}$, then multiple alignment does not exist and algorithm quits. Otherwise, proceed to step 6.
6. Recalculate $N-1$ pairwise alignments between structure x and other structures, with SSEs found in list $\mathcal{L}$ excluded from consideration, and return to step 3.

As seen from the above, the described algorithm may be implemented using any method for pairwise alignment. Using the similarity Q-score is an empirical element of the algorithm. The score was chosen on the ground of observation, described in Ref. [1], that it represents a considerably better measure for structural similarity than the more conventional r.m.s.d. and alignment length. For the pairwise alignments, we employ the SSM algorithm [1], being encouraged by its efficiency and quality quoted recently in an independent study [22]. Because SSM algorithm is based on matching SSEs, it allows for efficient removal of non-matching SSEs from consideration in step 6 above.

3.2 Multiple C_α alignment

Multiple alignment of structural elements yields a list of geometrically equivalent SSEs in the given structures. These data can be used for identifying common substructures in general and may be sufficient in some studies. A detail analysis of structural similarity requires structure alignment on the level of individual residues, including those not contained in SSEs. Below we describe an algorithm for multiple alignment of residues represented by their C_α atoms.

The algorithm follows the ideas of SSM algorithm for pairwise C_α alignment (SSM-PA), described in Ref. [1]. Using SSE alignment as an initial guess for the superposition of structures, SSM algorithm looks for pairs of C_α atoms which may be mapped onto each other such as to maximise a score function. Obtained alignment is then used for the calculation of improved superposition and the whole process is iterated until alignment does not change.

The SSM-PA algorithm may be adapted to multiple alignment after corresponding changes in its part that maps C_α atoms and redefinition of the score function. Below we discuss these changes and summarise the algorithm. In what follows, a_i stands for ith C_α atom of chain A and $|a_i, b_j|$ is distance between two atoms. We will also refer to groups of atoms, all from different chains, as $\mathcal{G}_{i,j,k\ldots} = \{a_i, b_j, c_k \ldots\}$. A group is considered as mapped, if all atoms in the group are found to be in geometrically equivalent positions.

SSM-PA defines a pair of atoms (a_i, b_j) as mappable if they belong to compatible SSEs (see details in Ref. [1]) and $|a_i, b_j| \leq |a_i, b_m|$ and $|a_i, b_j| \leq |a_n, b_j|$ for any unmapped atoms a_n and b_m. This definition allows to identify the pair unambiguously and efficiently. Having sorted all pairs by increasing distance prior the mapping, SSM-PA builds optimal C_α alignment by mapping pairs one-by-one starting from top of the list. Each new pair is checked for the connectivity conflict with all previously mapped pairs, that is, for any two mapped pairs (a_i, b_j) and (a_n, b_m) the equality $\text{sign}(n - i) = \text{sign}(m - j)$ should hold.

In order to find mappable atoms in more than two chains, one has to define a distance measure for groups of atoms, $|\mathcal{G}|$. A few distance measures may be proposed, for example,

$$|\mathcal{G}| = \sqrt{\frac{1}{2N(N-1)} \sum_{x,y \in \mathcal{G}} |x, y|^2} \tag{2}$$

$$|\mathcal{G}| = \max_{x,y \in \mathcal{G}} |x, y| \tag{3}$$

$$|\mathcal{G}| = \max_{x \in \mathcal{G}} |x, \bar{g}| \tag{4}$$

where $\bar{g}$ is a central-mass atom in the group. One may see that a straightforward use of these or similar distance measures for mapping groups of atoms results in the evaluation of a large number of groups, even after introducing a reasonable distance cut-off. It may be shown that computation complexity of such algorithm is proportional to $N!$, which makes it unfeasible for $N > 5 - 8$ structures. Therefore, we suggest a simplified procedure for the identification of mappable groups of atoms.

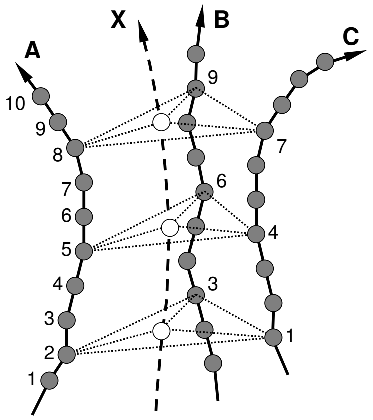

Fig. 2. Schematic of multiple C_α alignment. A fragment of superposed chains is shown in the Figure. Mapped groups of atoms are connected by dotted lines, chain X represents the consensus structure. Suppose that structure B is the closest one to X in pairwise score, then mapped groups $\{a_2, b_3, c_1\}$, $\{a_5, b_6, c_4\}$ and $\{a_8, b_9, c_7\}$ are identified as atoms b_3, b_6, b_9 and atoms from chains A and C closest to them. See text for details.

Introduce *consensus* structure X made of atoms placed in mass centers of the mapped groups (cf. Fig. 3.2). Next, find structure A^* that is closest to X in pairwise score (initially this structure may be defined as one with minimal sum of pairwise scores to other structures). Now one can identify mappable groups as those made from atoms a_i^* and atoms mappable to them in all other structures, chosen as in pairwise SSM alignment procedure [1], outlined above.

The proposed approach may be viewed as a simplified version of the central star method used in multiple sequence alignment (cf. Ref. [23]). While sequence alignment may be done in one pass, structure alignment involves recalculation of structure superposition after each alignment, which recalculation may change the choice of structure A^*. We found in a number of trial studies that this approach is a good approximation to the full-metric solution. Both approaches give identical answers for the alignment of structures with pronounced similarity, and a moderate number of differences (few percent of aligned residues) in case of dissimilar structures.

As noted in Ref. [1], not all mappings improve the alignment score. After all possible mappings are done, the algorithm should try to improve the alignment score by unmapping the groups with large distance measure $|\mathcal{G}|$. We define the alignment score as

$$Q = N_{align}^2 / \left\{ \left[1 + (D_{\mathcal{G}}/R_0)^2 \right] N_{min} N_{max} \right\} \tag{5}$$

where N_{align} is number of aligned groups, N_{min} and N_{max} are minimal and maximal number of residues in the aligned chains, R_0 is the same empirical parameter as in Eq. (1). $D_{\mathcal{G}}$ is calculated as r.m.s.d. of all mapped groups:

$$D_{\mathcal{G}} = \sqrt{\sum_{a_i^*} |\mathcal{G}_{\ldots i \ldots}|^2 / N_{align}} \tag{6}$$

where any of Eqs. (2-4) may be used for the calculation of $|\mathcal{G}_{\ldots i \ldots}|$. We use Eq. (2) because then Eq. (5) reduces to the pairwise Q-score [1] at number of structures $N = 2$.

Algorithm of multiple C_α alignment

1. Using the results of multiple SSE alignment, make initial superposition of structures and find structure A^* with least sum of pairwise Q-scores to other structures.
2. Calculate core C_α-alignment as an intersection of all pairwise alignment obtained in the last iteration of multiple SSE alignment.
3. Identify all mappable groups of atoms respecting to umapped atoms a_i^* as described above, and sort them by increasing the distance score $|\mathcal{G}_{\ldots i \ldots}|$. Starting from top of the list, map groups that do not have the connectivity conflict with all previously mapped groups.
4. Unmap groups in the reverse order until maximum value of Q-score, as defined by Eqs. (5,6), is reached.
5. Mapped groups of atoms represent a multiple alignment. If it does not differ from the one previously obtained then quit. Otherwise proceed to step 6.
6. Calculate consensus structure as mass centers of the mapped groups (see Fig. 3.2). Using algorithm for fast optimal superposition, described in Ref. [1], superpose all structures with the consensus structure.
7. Identify structure A^* which superposes with best pairwise score on the consensus structure, and proceed to step 2.

3.3 Implementation, output data and scores

The described algorithm of multiple alignment of protein structures in three dimensions has been implemented as an additional function of the EBI-MSD web-server SSM, which also may be used as a standalone (off-line) application in in-house setups. The development is based on the new CCP4 Coordinate Library [24]. The output data include:

Alignment length: number of aligned groups of C_α atoms
Consensus r.m.s.d. and Q-score: r.m.s.d. and Q-scores of each structure alignment to consensus structure
Overall r.m.s.d. and Q-score: calculated as Eqs. (6) and (5), respectively
Superposition matrices: rotation-translation matrices of best structure superposition on consensus structure
Pairwise scores: $N \times N$ matrices of pairwise r.m.s.d., Q-score and sequence identity
SSE and C_α alignments: tables of aligned SSEs and residues.

All output data may be downloaded in XML or plain text format (and FASTA format for aligned sequences), superposed structures may be visualised using the Rasmol [25] software.

4 Results and discussion

Fig. 4 represents data on the computational performance of the described algorithm, obtained from the log of SSM server at EBI-MSD for 2004-2005 year

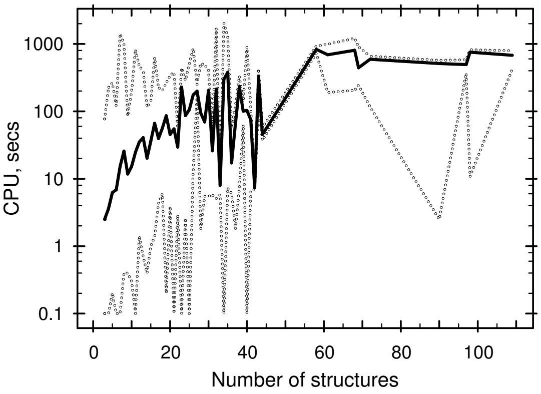

Fig. 3. CPU time as a function of the number of aligned structures obtained from the log of SSM server at EBI-MSD. All calculations were done on a single 1.2Ghz PC. Upper dotted line represents the maximum CPU time required, lower dotted line - minimum CPU time, and solid line gives the average. 95% of the data correspond to multiple alignment of up to 30 structures.

period. As may be seen from the Figure, calculation time is not a simple function of the number of aligned structures N. However, in the region of $3 \leq N \leq 30$, where most of the data have been collected, the average computation time has polynomial trend on N. In each particular case, calculation time also depends on the structure size (number of SSEs) and structural similarity: calculations are, on average, longer for larger and less similar structures. As may be seen from Fig. 4, these factors make a difference of more than 4 orders of magnitude.

The computational complexity of MA algorithm may be estimated as $O(N^2 n_m)$ times complexity of pairwise alignment, where n_m stands for the number of SSEs in the longest chain. Complexity of SSM-PA depends on structure topology and similarity and ranges from $O(nm)$ to $O(m^{n+1}n)$, where n, m are the numbers of SSEs in the aligned structures.

Figs. 4A,B present the typical results of multiple alignment. As seen from Fig. 4A, our MA algorithm is capable of discovering common substructures in different-fold structures, as defined by SCOP classification [28]. The β-sheet, common to all structures, was aligned with overall r.m.s.d. of 2.7Å and Q-score of 0.14, which implies a noticeable similarity. This similarity is present despite a rather low sequence identity of the aligned parts, which ranges from 0 for pair `1sar:A-1jqq:C` to 0.14 for pair `1sar:A-1jy4:B`.

Multiple alignment of same-family structures usually shows high structural similarity, as one would expect to obtain from SCOP classification. Fig. 4B demonstrates very clearly that structural differences occur only on protein surface, while internal parts match closely, forming a core of chain fold. In the example in Fig. 4B, aligned parts were matched with overall r.m.s.d. of 1.55Å and Q-score of 0.53, which implies a strong structural similarity. Sequence identity of the aligned structures in this example varies from 0.31 (`4dfr:A-1dhf:A`) to 1.0 (`1ra8-5dfr`).

Since there is no commonly accepted mathematical definition for multiple structure alignment, quality assessment of the results is difficult. A detail discussion of this question is outside the scope of present study. Table 1 shows a

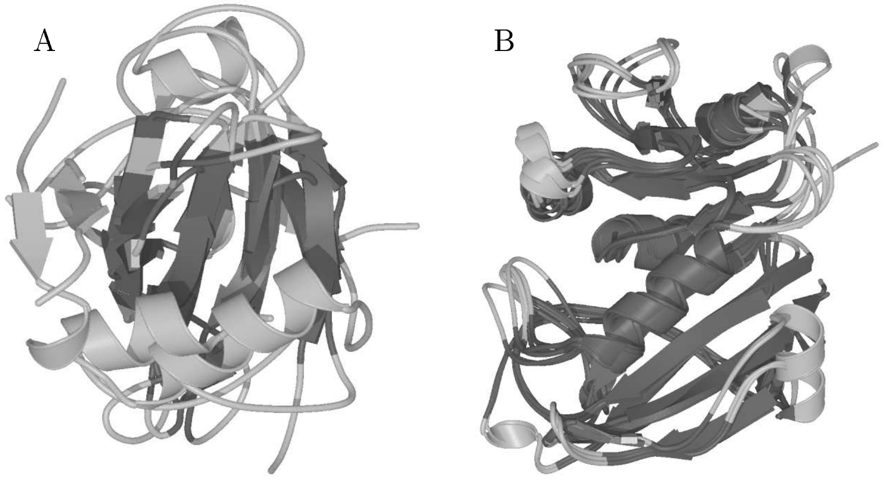

Fig. 4. Results of multiple alignment of A) different-fold structures 1sar:A, 1lqm:B, 1jqq:C and 1jy4:B (SCOP families d.1.1.2, d.17.5.1, b.34.2.1 and k.35.1.1, respectively), and B) same-family structures 4dfr:A, 1ra8, 5dfr, 1dhf:A, 1mvs:A, 1ia1:B and 1ia3:A (all belong to SCOP family c.71.1.1). Aligned parts are shown in dark grey. The pictures were obtained using Molscript [26] and Raster 3D [27] software.

typical example of comparison of multiple alignments obtained from Combinatorial Extension [21], MASS [20] and SSM servers.

Visual inspection of the alignments reveals that the servers, in general, agree with each other. As seen from Table 1, SSM's alignments are somewhat longer than those from MASS at higher r.m.s.d. (alignment length in CE seems to be reported wrongly, see remarks in the Table caption). This fact means that, comparing to SSM, MASS is more willing to sacrify the alignment length in favour of lower r.m.s.d. The balance between N_{align} and $RMSD$ depends on empirical parameters (such as distance cut-off) used in particular algorithms, and, generally, is not an indicator of a method's quality or robustness. We discussed this question in details in Ref. [1].

Comparison of the servers performance may be done only with the following important remarks. Firstly, the run time depends drastically on the selection of aligned structures, which is demonstrated in Fig. 4 by a considerable difference between the maximal and minimal CPU time required for the alignment. Therefore, fair comparison may be done only using the averaged run times from the servers' logs, which are not available on-line. Secondly, in difference of SSM, CE-MA and MASS are not interactive servers. Instead, they deliver results by e-mail. We measured the response time of CE-MA as a difference between the "send" time tags of the e-mails confirming the submission and delivering the results. MASS does not confirm submission by e-mail, and we measured its response time as a difference between the actual time of delivery and delivery time for a MA of 3 identical structures, which is supposed to be very fast. All mea-

N	CE		$MASS$		SSM	
	N_{align}	$RMSD$	N_{align}	$RMSD$	N_{align}	$RMSD^{\ddagger}$
8	205*	1.2	130	1.1	146	1.5
12	183*	1.4	121	1.1	143	1.6
16	188*	1.5	118	1.1	140	1.5
18	187*	1.4	119	1.1	140	1.5
20	187*	1.4	118	1.1	140	1.5
24	187*†	1.4	39	1.1	77	1.5

Table 1. Alignment lengths and r.m.s.d. of multiple alignments obtained from CE-MA [21], MASS [20] and SSM servers (present study). The initial set of 24 structures contained PDB entries `4dfr:A`, `1dyh:A`, `1dyi:A`, `1rb3:A`, `2drc:B`, `1ra3`, `1re7:A`, `1ra9`, `1rx2`, `5dfr`, `1dg8:A`, `1dg5:A`, `1dhf:A`, `1u70:A`, `1dr2`, `1hfq`, `1u72:A`, `1pd9:A`, `1dyr`, `1j3j:B`, `1ia4:B`, `1vj3:A`, `1m78:A` and `1t6t:2`. The entries were picked from the results of pairwise alignment of `4dfr:A` to all entries of PDB such that Q covers a range of 0.2 to 1. Then the subsets of 8, 12, 16, 18 and 20 structures were obtained by leaving every 3^{rd}, and removing every 2^{nd}, 3^{rd}, 4^{th} and 6^{th} structure from the set, respectively. *Alignment length, reported by CE, is apparently wrong because it exceeds chain lengths of individual structures (160 for `4dfr:A`). † In 24-structure set, CE-MA omitted PDB entry `1t6t:2`. ‡ Consensus r.m.s.d. is shown.

surements were done in off-peak time period, without parallel submissions. The last factor, that affects the comparison, is the server's hardware. SSM-MA runs on a single 1.2Ghz Linux PC, and it is not likely that it may get a substantial advantage, if any, on the hardware basis.

Figure 4 shows comparison of response time, measured as described above, obtained from CE-MA, MASS and SSM for producing multiple alignments in Table 1. As seen from the Figure, SSM outperforms CE-MA by almost an order of magnitude for all data sets in Table 1. MASS seems to be 4 to 6 times slower than SSM except for the data set of 8 structures, when MASS is as fast as SSM.

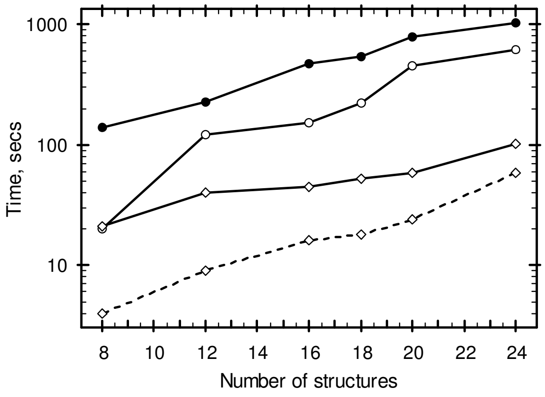

Fig. 5. Response time of CE-MA [21] (filled circles), MASS [20] (open circles) and SSM (dimonds) for producing alignments in Table 1. Dashed line shows CPU time of SSM. See text for details.

5 Conclusion

We have described here the algorithm of multiple alignment of protein structures employed in the EBI-MSD web service SSM. The service has been launched in June 2002 and since then served tens of thousands requests yearly. Vast experience of using SSM proved its high efficiency and quality of the results [22]. Our MA algorithm is different from a few others avaliable in that it seeks a solution by gradual removal of structural elements that are less likely to get aligned, rather than by a progressive clustering of the most similar chains. We have shown in this paper that SSM-MA is capable to handle large sets of structures and in most instances the results are delivered in a few minutes time. We have also described the basic scores used in SSM-MA output. These scores are derived from those of pairwise structure alignment by generalisation on the many-structure case. Like in the case of PA (cf. Ref. [1]), our experience suggests that Q-score is a better measure of structural similarity than the traditionally used r.m.s.d. and alignment length. As found, SSM-MA is capable of picking similarities in remote structures from different SCOP folds and classes, which suggests usability of the method for structure classification and studying the structure-function relationships.

Acknowledgement
E.K. is supported by the research grant No. 721/B19544 from the Biotechnology and Biological Sciences Research Council (BBSRC) UK.

References

1. Krissinel, E. and Henrick, K.: Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Cryst. **D60** (2004) 2256—2268.
2. Chotia, C. and Lesk, A.M.: The relation between the divergence of sequence and structure in proteins. EMBO J. **5** (1986) 823–826.
3. Chotia, C.: One thousand families for the molecular biologist. Nature **357** (1992) 543–544.
4. Hubbard, T.J.P. and Blundell, T.L.: Comparison of solvent-inaccessible cores of homologous proteins – definitions useful for protein modelling. Protein Engng. **1** (1987) 159–171.
5. Holm, L. and Sander, C.: Protein structure comparison by alignment of distance matrices. J. Mol. Biol. **233** (1993) 123-138.
6. Orengo, C.A. and Taylor, W.R.: SSAP: Sequential Structure Alignment Program for protein structure comparison. Meth. Enzym. **266** (1996) 617-635.
7. Falicov, A. and Cohen, F.E.: A surface of minimum metric for the structural comparison of proteins. J. Mol. Biol. **258** (1996) 871-892.
8. Gerstein, M. and Levitt, M.: Using iterative dynamic programming to obtain accurate pairwise and multiple alignments of protein structures. In Proc. of the Fourth Int. Conf. Intell. Syst. Mol. Biol., Menlo Park, Calif.: (1996) AAAI Press, pp. 59-67.
9. Singh, A.P. and Brutlag, D.L.: Hierarchical protein structure superposition using both secondary structure and atomic representations. In Proc. Int. Conf. Intell. Syst. Mol. Biol. ISMB-97: (1997) AAAI Press, pp. 284-293.

10. Vriend, G. and Sander, C.: Detection of common three-dimensional substructures in proteins. Proteins **11** (1991) 52-58.
11. Mizuguchi, K. and Go, N.: Comparison of spatial arrangements of secondary structural elements in proteins. Protein Engng. **8(4)** (1995) 353-362.
12. Mitchell, E.M., Artymiuk, P.J., Rice, D.W. and Willett, P.: Use of techniques derived from graph theory to compare secondary structure motifs in proteins. J. Mol. Biol. **212** (1990) 151-166.
13. Alexandrov, N.N.: SARFing the PDB. Protein Engng. **9** (1996) 727-732.
14. Grindley, H.M., Artymiuk, P.J., Rice, D.W. and Willett, P.: Identification of tertiary structure resemblance in proteins using a maximal common subgraph isomorphism algorithm. J. Mol. Biol. **229** (1993) 707-721.
15. Shindyalov, I.N. and Bourne, P.E.: Protein structure alignment by incremental combinatorial extension (CE) of the optimal path. Protein Engng. **11(9)** (1998) 739-747.
16. Gibrat, J.-F., Madej, T. and Bryant, S.H.: Surprising similarities in structure comparison. Current Opinion in Structural Biology **6** (1996) 377-385.
17. Kleywegt, G.J. and Jones, T.A.: Detecting folding motifs and similarities in protein structures. Meth. Enzym. **277** (1997) 525-545.
18. Russell, R.B. and Barton, G.J.: Multiple protein sequence alignment from tertiary structure comparison. Proteins: Struct. Funct. Genet. **14** (1992) 309–323.
19. Shatsky, M., Nussinov, R. and Wolfson, H.J.: MultiProt - a Multiple Protein Structural Alignment Algorithm. In Lecture Notes in Computer Science: (2002) Springer Verlag, pp. 2452:235–250.
20. Dror 0., Benyamini H., Nussinov R. and H. Wolfson: Multiple structural alignment by secondary structures: algorithm and applications. Protein Science **12** (2003) 2492–2507.
21. Guda C., Lu S., Scheeff E.D., Bourne P.E. and Shindyalov I.N.: CE-MC: a multiple protein structure alignment server. Nucl. Acids Res. **32** (2004) W100–W103.
22. Kolodny, R., Koehl, P. and Levitt, M.: Comprehensive Evaluation of Protein Structure Alignment Methods: Scoring by Geometric Measures. J. Mo. Biol. **346** (2005) 1173–1188.
23. Gusfield, D. *Algorithms on Strings, Trees and Sequences.* Cambridge University Press, New York, (1997), pp 348–350.
24. Krissinel, E.B., Winn, M.D., Ballard, C.C., Ashton, A.W., Patel, P., Potterton, E.A., McNicholas, S.J., Cowtan, K.D. and Emsley, P.: The new CCP4 Coordinate Library as a toolkit for the design of coordinate-related applications in protein crystallography. Acta Cryst. **D60** (2004) 2250—2255.
25. Sayle, R. A., and Milner-White, E. J.: RasMol: Biomolecular graphics for all. Trends in Biochemical Sci. **20** (1995) 374-376.
26. Kraulis, P.J.: MOLSCRIPT: a program to produce both detailed and schematic plots of protein structures. J. Appl. Cryst. **24** (1991) 946-950.
27. Merritt, E.A. and Bacon, D.J.: Raster3D: Photorealistic Molecular Graphics. Meth. Enzymol. **277** (1997) 505-524.
28. Murzin, A.G., Brenner, S.E., Hubbard, T. and Chothia, C.: SCOP: a structural classification of proteins database for the investigation of sequences and structures. J. Mol. Biol. **247** (1995) 536-540.