BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 17 2013, pages 2112–2121
doi:10.1093/bioinformatics/btt360

Sequence analysis

Advance Access publication June 21, 2013

Optimizing multiple sequence alignments using a genetic algorithm based on three objectives: structural information, non-gaps percentage and totally conserved columns

Francisco M. Ortuño[1,*], Olga Valenzuela[2], Fernando Rojas[1], Hector Pomares[1], Javier P. Florido[3], Jose M. Urquiza[4] and Ignacio Rojas[1]

[1]Department of Computer Architecture and Computer Technology, CITIC-UGR, [2]Department of Applied Mathematics, University of Granada, 18071 Granada, Spain, [3]Bioinformatics Department, Genomics and Bioinformatics Platform of Andalusia (GBPA), 41092 Seville, Spain and [4]Chromatin and Disease Group, Bellvitge Biomedical Research Institute (IDIBELL), L'Hospitalet, Barcelona 08907, Spain

Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: Multiple sequence alignments (MSAs) are widely used approaches in bioinformatics to carry out other tasks such as structure predictions, biological function analyses or phylogenetic modeling. However, current tools usually provide partially optimal alignments, as each one is focused on specific biological features. Thus, the same set of sequences can produce different alignments, above all when sequences are less similar. Consequently, researchers and biologists do not agree about which is the most suitable way to evaluate MSAs. Recent evaluations tend to use more complex scores including further biological features. Among them, 3D structures are increasingly being used to evaluate alignments. Because structures are more conserved in proteins than sequences, scores with structural information are better suited to evaluate more distant relationships between sequences.

Results: The proposed multiobjective algorithm, based on the non-dominated sorting genetic algorithm, aims to jointly optimize three objectives: STRIKE score, non-gaps percentage and totally conserved columns. It was significantly assessed on the BAliBASE benchmark according to the Kruskal–Wallis test ($P<0.01$). This algorithm also outperforms other aligners, such as ClustalW, Multiple Sequence Alignment Genetic Algorithm (MSA-GA), PRRP, DIALIGN, Hidden Markov Model Training (HMMT), Pattern-Induced Multi-sequence Alignment (PIMA), MULTIALIGN, Sequence Alignment Genetic Algorithm (SAGA), PILEUP, Rubber Band Technique Genetic Algorithm (RBT-GA) and Vertical Decomposition Genetic Algorithm (VDGA), according to the Wilcoxon signed-rank test ($P<0.05$), whereas it shows results not significantly different to 3D-COFFEE ($P>0.05$) with the advantage of being able to use less structures. Structural information is included within the objective function to evaluate more accurately the obtained alignments.

Availability: The source code is available at http://www.ugr.es/~fortuno/MOSAStrE/MO-SAStrE.zip.

Contact: fortuno@ugr.es

Supplementary Information: Supplementary material is available at *Bioinformatics* online.

*To whom correspondence should be addressed.

Received on October 10, 2012; revised on May 24, 2013; accepted on June 18, 2013

1 INTRODUCTION

Multiple sequence alignments (MSAs) are widely used strategies in current molecular biology. These approaches are often used for homology transfer (Doolittle, 1981; Fitch, 1966), where poorly characterized sequences are compared with well-studied homologs from typical model organisms. MSA strategies have traditionally been applied to researches in phylogenetic analyses, structural modeling, functional predictions or sequence database searching (Bacon and Anderson, 1986). MSA tools have also been implemented in applications to predict protein structures and interactions (Chou and Fasman, 1978; Taylor and Thornton, 1984), mutations (Schneider *et al.*, 1986) or to reconstruct phylogenetic trees (Feng and Doolittle, 1987). The development of novel experimental techniques, such as next-generation sequencing and high-throughput experiments, has prompted a great demand of MSA tools. Because these techniques provide mainly new nucleotide sequences and their subsequent products, MSA tools usually help to extract biological meanings from such information. Current MSA tools are capable of dealing with and efficiently analyze the massive amount of information generated by these former techniques by using advanced computational approaches based on well-known artificial intelligence and machine-learning algorithms (hidden Markov models (HMMs), support vector machines, etc). Besides, MSA methodologies also take advantage of functional, structural and genomic information to obtain more accurate alignments in a reasonable time (Kemena and Notredame, 2009). Taking all these ideas into consideration, MSAs are becoming one of the most powerful and essential procedures of analysis in bioinformatics (Li and Homer, 2010).

Traditionally, several strategies have been applied to align multiple sequences, mainly classified as progressive algorithms (Hogeweg and Hesper, 1984) or consistency-based methods (Gotoh, 1990). Both approaches were also combined with other relevant computational strategies to obtain more accurate alignments. Recently, more sophisticated tools in MSA have

included additional data referring to proteins (domains, structures or homologies) to align sequences (O'Sullivan *et al*., 2004; Pei and Grishin, 2007). Such additional features enrich the alignment information building more realistic solutions. However, the consumed time is excessive and improvements are just relevant in specific cases with less related sequences. Moreover, these methods can be run when additional features are unavailable or unknown, though they could provide inefficient alignments.

Genetic algorithms (GAs) are also widely used to build MSAs. GAs are helpful in MSA because they can be implemented independently of the objective function (Naznin *et al*., 2011). Thus, GA algorithm can define multiple evaluations regardless of any modification in the optimization procedure. GAs can also be easily parallelized to significantly reduce the computational time. Consequently, several methodologies, such as SAGA (Notredame and Higgins, 1996), MSA-GA (Gondro and Kinghorn, 2007), RBT-GA (Taheri and Zomaya, 2009) or VDGA (Naznin *et al*., 2011), have already applied GAs to build MSAs.

Although there are many MSA methodologies, they usually achieve different solutions for the same set of sequences because each strategy is focused on specific biological features. Consequently, there is no consensus about which method builds more accurate alignments (Nuin *et al*., 2006; Sierk *et al*., 2010). Besides, these MSA tools could achieve suboptimal solutions where specific regions within the alignments are more accurate than others depending on the biological features found at these particular regions. These divergences have also a negative influence on subsequent phylogenetic analyses, as wrong phylogenetic trees are obtained when alignments are inaccurate (Wong *et al*., 2008). For this reason, some other methods (Redelings and Suchard, 2005; Ronquist and Huelsenbeck, 2003) take advantage of jointly optimizing both phylogenetic trees and alignments. These methods aim to avoid the bias generated by guide trees in progressive methods, though they do not still achieve good performances in terms of structure. Therefore, the choice of the most suitable aligner is an essential problem, which has not been completely solved yet.

Another challenge in MSA is to provide an efficient evaluation method to measure the alignment accuracy. MSA strategies have usually applied well-known matrices, such as point accepted mutation (PAM) (Dayhoff *et al*., 1978) or BLOSUM (Henikoff and Henikoff, 1992), which only consider nucleotide or amino acid information to evaluate every aligned pair of residues. However, when the number of sequences increases or longer and more distant sequences are included, alignments are more likely to be inaccurate using such scores (Liu *et al*., 2009). In these cases, additional information is necessary to complement alignment evaluations. Therefore, current scores are increasingly using supplementary information, such as homologies or protein structures. Thus, some approaches can benefit from homology profiles provided, e.g. by PSI-BLAST (Altschul *et al*., 1997), to evaluate alignments. Additionally, as structures are evolutionarily more conserved than sequences in proteins, structural information also provides more distant relationships between sequences (Kemena and Notredame, 2009). For instance, Kececioglu *et al*. (2010) provided a novel scoring scheme to evaluate MSAs from their predicted secondary structures. Other scores, such as contact accepted mutation (Lin *et al*., 2003) and STRIKE (Kemena *et al*., 2011) scores also estimated the molecular contacts from protein structures to calculate alignment accuracies.

In this article, a novel multiobjective genetic approach has been developed. This method is named ***M**ultiobjective **O**ptimizer for **S**equence **A**lignments based on **Str**uctural **E**valuations* (MO-SAStrE). It takes advantage of three objectives that are used to evaluate alignments generated by the GA: STRIKE score (Kemena *et al*., 2011), totally conserved (TC) columns and percentage of non-gaps. Alignments are first coded in a novel representation, which is useful for applying efficient mutation and crossover operators. This algorithm is implemented through the well-known multiobjective non-dominated sorting genetic algorithm (NSGA-II) approach. It is assessed by the BAliBASE benchmark v3.0 (Thompson *et al*., 2005). Alignments from MO-SAStrE are finally compared with results shown by other known genetic and non-genetic alignment algorithms.

2 METHODS

2.1 Input sequence dataset

The proposed multiobjective algorithm must be tested through a dataset defined by several input sequences. The BAliBASE dataset (v3.0) (Thompson *et al*., 2005) defines a well-known benchmark to standardize the comparison of sequence alignment results. It consists of a group of protein sequences that are properly prepared to be aligned by MSA algorithms. The dataset includes 218 sets of sequences, which were manually extracted from the protein data bank (PDB) (Berman *et al*., 2000). It is organized in five reference subsets, named References or Ref., according to their sequence families or similarities. The first reference subset (Ref.1) is separated in two versions (Ref.1 v.1 and Ref.1 v.2). The first version (Ref.1 v.1) includes less similar sequences, which are interesting because they are more difficult to be accurately aligned.

BAliBASE also provides a set of handmade alignments (*gold standard*) to evaluate alignments obtained by other tools. Thus, this benchmark calculates BAliscore, a standard Sum-of-Pairs score to evaluate alignments compared with their gold standard. Here, the BAliscore evaluation was used to compare the MO-SAStrE performance against other similar MSA methodologies.

2.2 Alignment approaches

In this article, eight representative MSA tools were selected to obtain initial alignments. Both progressive and consistency-based methods were included in these representative tools (see a summary in the Supplementary Table S1). Among progressive algorithms, ClustalW (Thompson *et al*., 1994), Muscle (Edgar, 2004), Kalign (Lassmann and Sonnhammer, 2005), Mafft (Katoh *et al*., 2002) and RetAlign (Szabo *et al*., 2010) were chosen in the proposed optimization. ClustalW designs a clustering tree algorithm to find the final alignment through a distance score matrix and a gap weighting scheme. Muscle develops a three-stage strategy to refine alignments and to align faster. Kalign uses the Wu-Manber string-matching algorithm to improve the measurement of distances within a classical progressive approach. Mafft reduces the computational cost by identifying common homologies through the fast Fourier transform. Finally, RetAlign applies a progressive corner cutting algorithm to identify optimal and suboptimal alignments in a network.

Besides, three additional algorithms based on consistency were also included in the optimization: T-Coffee (Notredame *et al*., 2000), fast statistical alignment (FSA) (Bradley *et al*., 2009) and ProbCons (Do *et al*., 2005). T-Coffee stores in a library the number of times each pair of residues matched in previously built pairwise alignments. T-Coffee also

evaluates such pairwise alignments in regard to third sequences. The FSA compares pairwise alignments by a statistical analysis framework. FSA estimates the insertion and deletion processes through a pair of HMMs. This method provides a faster procedure, but it usually achieves less accurate alignments owing to an excessive number of gaps. Finally, ProbCons also includes HMMs to optimize the classical scoring schemes. It applies a biphasic penalty procedure to penalize gaps and mismatches in alignments. Methods based on HMM profiles, such as FSA or ProbCons, usually outperform other alignment methods, especially in terms of the structure-superposition quality (Kemena and Notredame, 2009). The 218 sets of sequences proposed by BAliBASE were then aligned using these eight programs. All of them were run with their default parameters, though they can be modified according to the user preferences. These specific initial alignments were chosen because they were quickly obtained and might be improved. In case more accurate initial alignments were provided, MO-SAStrE could even return better output alignments. Moreover, the efficiency of including previously obtained solutions to build the initial set in GAs has widely been shown in the literature (Dasgupta *et al.*, 2009; Tsujimoto *et al.*, 2009).

2.3 Multiobjective algorithm

MSAs can be defined as multiobjective problems as there is no consensus about how alignments should be adequately evaluated and several features are currently being considered for this purpose. Additionally, including several suitable objectives provides more flexibility in the optimization procedure. Consequently, MO-SAStrE is implemented as a GA including three different evaluations: 3D structure, TC columns and gaps in alignments. The multiobjective approach is developed through the NSGA-II scheme (Deb *et al.*, 2002), as it is a classical and recognized method that produces efficient solutions. NSGA-II provides the subset of all optimal solutions, named Pareto front, by using the non-dominated sorting strategy. That is, the Pareto front includes those solutions that cannot be compared among them because there is no one that outperforms any other considering the three objectives. This feature is known as non-dominance relationship. Both Pareto front and dominance concepts are widely described in the Supplementary Material.

The MO-SAStrE procedure is designed as shown in Figure 1. First, alignments from input methodologies are included in the initial population. The coded alignments belonging to a population are called individuals. The population is then filled to N individuals (where N defines the population size) by using the crossover operator. Subsequently, the population is extended by the mutation and crossover operators, according to their assigned probabilities p_c and p_m, respectively (see the 'Operators' stage in Fig. 1). These operators are generally defined to build new individuals by combining already existing ones. The best individuals are then selected from the extended population to be included in the new generation. This selection is carried out progressively, taking the optimal non-dominated solutions (Pareto fronts) from the current population. If all individuals in the last included Pareto front (F_l in Fig. 1) cannot be added, they must be selected according to the crowding distance (see the 'Selection' subsection for details). Finally, when the total number of generations (G) is reached or the Pareto front does not change in consecutive generations, the optimal Pareto front in the last population is returned as the set of optimized alignments. This implementation of the NSGA-II approach was taken from the *Global Optimization* toolbox of ® Matlab (version R2010b). Individual codification, operators and fitness functions were own-designed for this specific purpose.

2.3.1 Codification Previous GAs in MSA tools (Notredame and Higgins, 1996; Taheri and Zomaya, 2009) coded alignments using the classical representation: the standard alphabet for amino acids and the '-' symbol for gaps (Fig. 2a). However, that representation could lead to more complex and inefficient operators. For this reason, a novel codification is proposed here. Alignments are represented as a matrix where

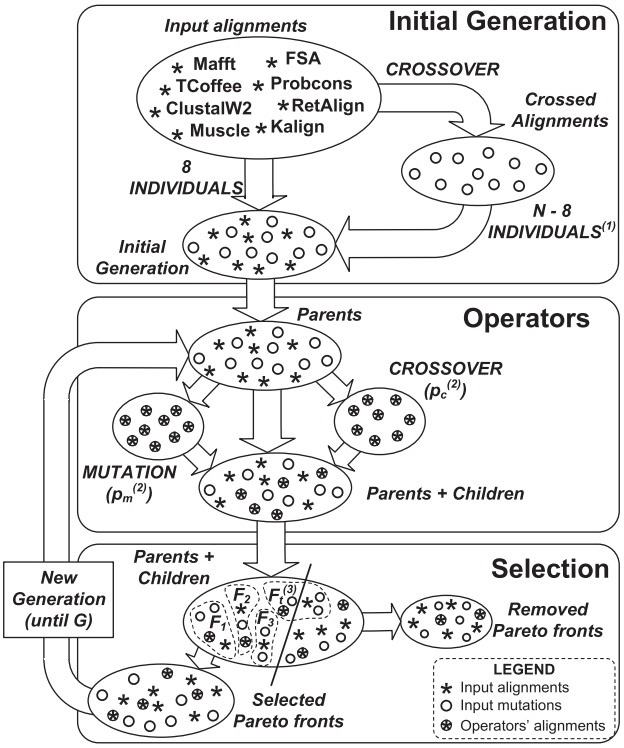

Fig. 1. MO-SAStrE flowchart. (1) The number of individuals in the population is defined as N. (2) p_c and p_m represent crossover and mutation probabilities, respectively. (3)F_l defines the last included Pareto front, where individuals must be selected according to the crowding distance

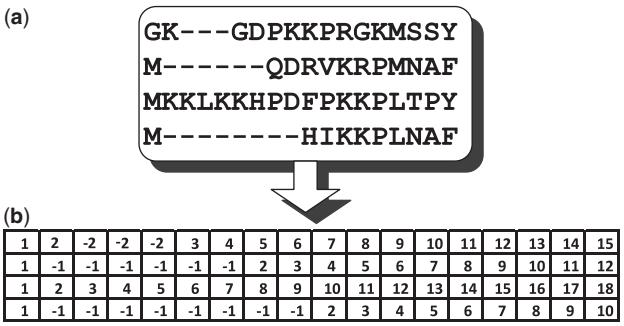

Fig. 2. Alignment codification in MO-SAStrE. (**a**) Standard representation of a MSA. (**b**) Alignment coded by a matrix of integer values: positions in their sequences (amino acids) and positions of the last amino acid denoted with a negative sign (gaps)

two conditions are fulfilled: (i) amino acids are coded by their positions in the sequence to which they belong; (ii) gaps are coded by the position of the last amino acid in the sequence where they belong, but with a negative value. Input alignments are coded before they are included in this tool. The whole optimization is done by using coded alignments (individuals). After the optimization ends, individuals are decoded and therefore returned to the standard alignment representation. An example of the proposed codification is shown in Figure 2. This representation aims to easily identify positions where the crossover operator will be applied. It avoids possible mistakes in the subsequent crossover performance, providing significant improvements in the alignment management (see details in the 'Operators' subsection).

2.3.2 Operators MO-SAStrE includes the two standard operators in GAs: mutation and crossover. These operations are applied to a subset of randomly chosen alignments from the population according to the probabilities p_m and p_c, respectively. They can then include new alignments not considered before. These operators are run for each generation in the optimization procedure. Because sequences in alignments cannot be altered, some modifications must be introduced in the classical implementation of both operators.

The *mutation* operator only mutates gaps, to keep the order of amino acids. A random set of closed gaps are then shifted to another random position in the same sequence. Two important aspects are introduced with such definition: first, new variants of alignments not taken into account until now can be introduced; second, columns containing only gaps can be removed, thus reducing the number of gaps. A specific example of the mutation operation is shown in Figure 3.

The *crossover* operator is designed as a one-point crossover. Firstly, the algorithm randomly selects one column from one of the parents, splitting it into two blocks. The same selected positions from this column are also found in the second parent, but not necessarily in the same column. Finally, selected blocks are crossed between these two parents. To match blocks from both parents, those undefined positions are filled with gaps. Thus, it can be assured that the obtained children do not alter their sequences. The complete operation is graphically explained in Figure 4. The crossover operator is the most important issue in outperforming input methodologies. Because alignments can be more accurate in some sectors than in others, this operation is essential for the optimization purpose. Therefore, crossed children could assemble the best sections from different parents, providing a more accurate alignment.

2.3.3 Evaluation Because MO-SAStrE is designed as a multiobjective algorithm, three different scores are included to evaluate each alignment: STRIKE score, percentage of TC columns and percentage of non-gaps.

The **STRIKE score** (Kemena *et al*., 2011) is a novel index for calculating alignment accuracies by using at least one known structure. The structural information is retrieved from the PDB (Berman *et al*., 2000). According to such a structure, the contacts between amino acids in the sequence are estimated. For the remaining sequences, the pairs of amino acids aligned in the same positions as the previously estimated contacts are retrieved. Such pairs of amino acids are then scored according to a novel scoring matrix provided by the STRIKE authors (Kemena *et al*., 2011). In case of several available structures, the STRIKE score is separately calculated for each structure and the averaged score is finally provided. This evaluation permits to identify the accuracy in the alignments better than other well-known scores such as BLOSUM (Henikoff and Henikoff, 1992) or PAM (Dayhoff *et al*., 1978). Moreover, the STRIKE score clearly outperforms the other evaluations when sequences are evolutionarily more distant. STRIKE score also shows a strong non-parametric correlation with the classical BAliscore. That is, both

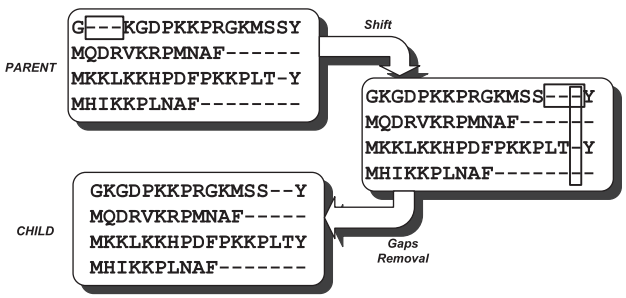

Fig. 3. Mutation procedure. Closed gaps are randomly chosen and shifted to another position. Full columns of gaps are then removed if they are found

BAliscore and STRIKE usually identify the same alignment as the best one when two different alignments are compared (in ~79% of cases) (Kemena *et al*., 2011).

The second fitness function, **the percentage of TC column**, takes into account the number of columns that are completely aligned with exactly the same amino acids. Some progressive methodologies usually favor partial alignment but not complete columns (Mirarab and Warnow, 2011). The number of complete columns is a widely accepted evaluation applied by several methodologies (Edgar, 2004; Thompson *et al*., 2005). Complete columns also indicate more conserved or special regions in sequences.

Finally, as commented above, some methodologies usually overuse gaps to increase identities in alignments (Nozaki and Bellgard, 2005). Thus, the third fitness function is measured as the number of amino acids in the sequences with respect to the number of gaps (**percentage of non-gaps**). Consequently, the proposed optimization tries to reduce the number of gaps, building more compact and realistic alignments.

Therefore, MO-SAStrE aims to optimize alignments according to a novel evaluation based on conserved structural information in sequences, but also reducing the number of gaps and keeping fully conserved sections. These three objectives must then be maximized to obtain more accurate alignments.

2.3.4 Selection The selection procedure is well-defined by the proposed NSGA-II algorithm (Deb *et al*., 2002). For each generation, the extended population (parents and children) is classified into different Pareto fronts to obtain a non-dominated sorting ($F_1, F_2, \ldots, F_t$ in Fig. 1). This procedure selects those individuals that are not outperformed by any other regarding the three objectives. Then, the best non-dominated Pareto front is progressively included within the next generation. Finally, when the new population is filled with the required number of individuals, the remaining Pareto fronts are discarded. A special case of this selection is the last considered front (F_t), as it is possible that only some individuals can be included within the next generation. In this case, NSGA-II proposed to include those individuals located in less explored areas or, in other words, distant individuals. The last individuals are then selected according to their distances to the nearest individuals. This measure is called the crowding distance (see formulation in the Supplementary Material).

2.4 Performance assessment

MO-SAStrE is defined as a stochastic procedure because the algorithm converges to different solutions when it is applied several times to the same problem. Consequently, several runs of the same problem must be carried out to statistically evaluate its performance. Zitzler *et al*. (2008) proposed several indicators to assess multiobjective stochastic optimizers: the hypervolume indicator (HV), dominance rankings or the attainment function method. The main goal of these quality indicators is to reduce the provided scores (three objectives) of multiple optimal solutions (Pareto front) to one single score, making the algorithm easier to assess. Thus, the HV (Zitzler *et al*., 2008) was selected to validate the optimization provided by the multiobjective algorithm (see formal definition of hypervolume in the Supplementary Material).

However, hypervolume is not the only strategy taken into account in the MO-SAStrE assessment. Because each problem was run several times, initial hypervolume values must also be compared with output hypervolume values provided by MO-SAStrE. Non-parametric tests are usually applied to validate these stochastic approaches (Conover, 1999). To assess this algorithm, the classical test proposed by Kruskal and Wallis (1952) is used. This Kruskal–Wallis test assesses whether there are significant differences between initial alignments and the optimized ones for independent repetitions in terms of the three objectives. For MO-SAStrE assessment, the initial and optimized hypervolume values are then compared.

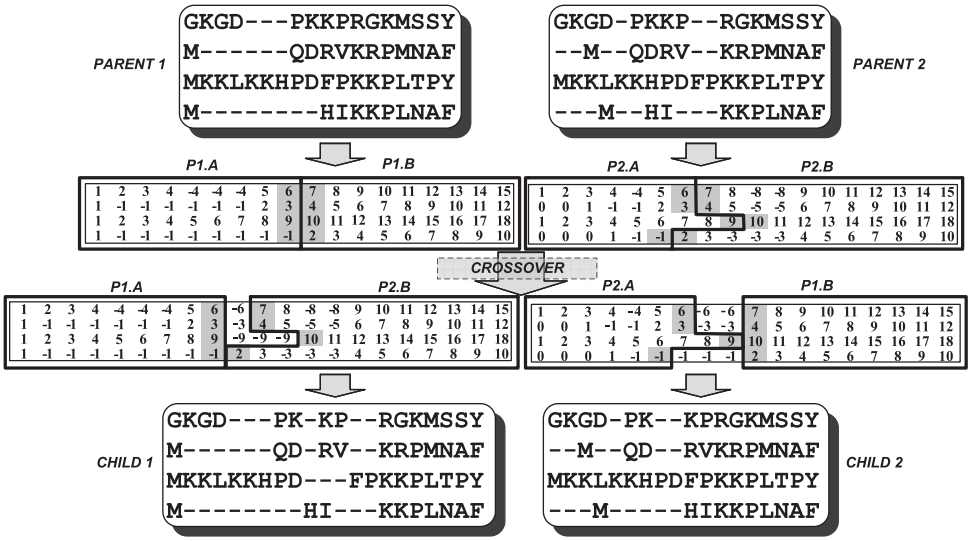

Fig. 4. Crossover operator. Both standard and novel codification are shown (see the codification procedure in Fig. 2). The first parent is divided into two blocks according to a selected column (P1.A and P1.B). Two blocks are also obtained from the second parent according to the same positions of the selected column (P2.A and P2.B). The crossed blocks are finally filled with gaps to match them

Finally, the performance of MO-SAStrE is compared with other genetic methods, namely SAGA (Notredame and Higgins, 1996), MSA-GA (Gondro and Kinghorn, 2007), RBT-GA (Taheri and Zomaya, 2009) and VDGA (Naznin *et al*., 2011). Other known non-genetic aligners are also included in these comparisons, namely ClustalW (Thompson *et al*., 1994), MultAlign (Barton and Sternberg, 1987), PIMA (Smith and Smith, 1992), PILEUP8 (Devereux *et al*., 1984), Dialign (Morgenstern *et al*., 1996), HMMT (Eddy, 1995), PRRP (Gotoh, 1996). 3D-COFFEE (O'Sullivan *et al*., 2004) was also included to compare MO-SAStrE against another aligner using structural information. To compare all of them, the authors in VDGA provided BAliscore results from 60 different problems in BAliBASE 2.0, which were also included in MSA-GA and RBT-GA publications. However, as BAliBASE 3.0 is applied here, a subset of 20 problems included in both versions of BAliBASE is taken. Then, MO-SAStrE is statistically compared for these 20 datasets through another non-parametric analysis, the Wilcoxon signed-rank test (Wilcoxon, 1945). The Wilcoxon test provides pairwise comparison between each two methods to validate if their mean ranks are significantly different. Therefore, it can be determined whether MO-SAStrE outperforms other similar tools.

3 RESULTS AND DISCUSSION

3.1 Selecting parameters

To configure the proposed multiobjective algorithm, five different parameters (population size, number of generations, probabilities of mutation and crossover and repetitions per problem) must be provided. These parameters were selected according to the standard values used by GAs (Eiben and Smith, 2008) (Supplementary Table S2). First, the population size was set to 100 alignments (individuals), as that same population size was also included in methods that are being compared, such as SAGA (Notredame and Higgins, 1996) or VDGA (Naznin *et al*., 2011).

On the other hand, although MO-SAStrE includes a stop condition, 500 generations were defined to assure the convergence and optimization of the alignments. Once these two parameters were set, the operator probabilities were determined. Because crossover is considered the main operator for this optimization, it is assumed that its probability must be the same or higher than the mutation's one. Consequently, the following pair of probabilities 80–20% was set for crossover and mutation, respectively. These probabilities values are a standard combination for GAs (Eiben and Smith, 2008). This parameter configuration was validated with a subset of 20 BAliBASE problems.

Finally, as the proposed optimizer is defined as a stochastic procedure, each problem must be run several times. In this case, each of the 218 problems was optimized 10 times. The same number of runs was also included in VDGA (Naznin *et al*., 2011) and RBT-GA (Taheri and Zomaya, 2009). A total of 2180 Pareto fronts were then obtained (10 solutions by 218 problems).

3.2 Optimization procedure

Firstly, the eight input alignments for each BAliBASE dataset were introduced in the MO-SAStrE algorithm. These alignments were progressively assembled into other optimized alignments as shown in the Supplementary Figure S3. Thus, the built solutions included partial alignments from previous methodologies and several gap shifts.

The multiobjective procedure returns the subset of non-dominated alignments (Pareto front). These obtained alignments are equally good and it is not possible to decide which one is more accurate according to the three objectives. Therefore, the selection of the best alignment only depends on the objective the users consider more useful regarding the specific aligned sequences. In case the alignment with the best STRIKE score was chosen, it would obtain more quality according to the sequence structures. In addition, those alignments with higher STRIKE scores are usually improved in terms of BAliscore (Kemena *et al*., 2011). Otherwise, whether the alignment with the highest percentage of non-gaps is selected, a more compact and realistic alignment could be obtained. Finally, a higher number of TC columns in

alignments provides a better quality in terms of the evolutionary homologies among sequences. For this reason, the Pareto fronts are hard to compare as multiple non-dominated solutions are considered and they should be assessed according to their three evaluations. For instance, Table 1 shows the objective values of MO-SAStrE alignments with regard to the eight inputs for the '1aab' problem in the Ref.1 v.1. In this case, the MO-SAStrE alignments outperform the input methodologies in at least one objective according to these evaluations. The same problem was also graphically compared through the initial and optimized Pareto fronts as shown in Figure 5. The optimized front achieves higher values in the three objectives than the initial one. The optimization procedure applied to the '1aab' problem was carried out for the 218 problems. Averages and standard deviations of the three objectives and the BAliBASE score (BAliscore) in the complete dataset are shown for each alignment methodology in the Supplementary Table S3. Similarly, the computing time taken by each methodology is depicted in the Supplementary Figure S4. Here, it can be observed that MO-SAStrE almost always achieves the best values in all the three objectives at the expense of a higher computing time. Despite these time differences, computing times obtained by MO-SAStrE are acceptable taking into account that the simplest and quickest input methodologies were chosen to be subsequently optimized. Additionally, it is shown in the Supplementary Table S3 that MO-SAStrE also outperforms the input methodologies in terms of the BAliscore.

3.3 Hypervolume analysis

To formally validate the results shown above, the HV was calculated as suggested by Zitzler *et al.* (2008) for multiobjective problems (see details in the 'Performance Assessment' subsection). Because the three independent objectives must be maximized here, better alignments lead to lower HV values (the higher objectives, the less covered space in HV). Previously, the three objectives were normalized to the range [0, 1] to give them the same weight. Then, HV could be interpreted as a measure of quality, which takes into consideration the three proposed objectives simultaneously. The HV was measured regarding a reference point (bounding point) (see Supplementary Material for details). Previously, it was assured that each problem and its HV value successfully converged to an optimized solution (the convergences of four different problems are shown in the Supplementary Figure S5). Subsequently, the HV values from the initial Pareto front (eight input alignments) can be compared with those obtained by MO-SAStrE. The HV comparisons were applied to the 218 problems, showing strictly better outcomes with MO-SAStrE in all of them. That is, the HV values obtained by MO-SAStrE from the 10 runs of each problem always outperformed the initial HV values. These results also showed that the optimized alignments achieved an average improvement of 63.01% according to HV indicators. Such an improvement even increases to 70.34% when sequences are less related and alignments become more difficult (Ref.1 v.1 subset in BAliBASE). A summary of the improvement associated to each subset in BAliBASE is shown in Table 2. Nevertheless, there were two problems where the improvements did not reach the 10%: '1aab' and '2trx' in Ref.2. Because these two datasets belong to a BAliBASE subset with higher similarity percentages, the initial alignments were already accurate. These specific problems also included some special features such as higher number of sequences or highly divergent lengths, making more difficult the optimization.

Table 1. Multiobjective scores for a specific problem

Method	STRIKE	Non-gaps (%)	TC (%)
ClustalW	2.4544	89.84	1.04
Muscle	2.6041	89.84	1.04
Kalign	2.4404	87.12	3.03
RetAlign	2.2210	79.13	2.75
Tcoffee	2.5116	89.84	1.04
ProbCons	2.5116	89.84	1.04
Mafft	2.3893	87.12	1.01
FSA	2.1857	69.00	0.80
BAliBASE Ref	2.5263	89.84	1.04
MO-SAStrE 1	2.4677	90.79	2.11
MO-SAStrE 2	2.6441	89.84	5.21
MO-SAStrE 3	2.6864	89.84	4.17
MO-SAStrE 4	3.0544	82.93	3.85
MO-SAStrE 5	3.1329	78.41	2.73

Note: Results are shown for the '1aab' dataset in Ref.1 v.1. Evaluations for input alignments and for MO-SAStrE alignments are represented. Although MO-SAStrE returned 30 alignments from the optimal Pareto front, five of them are shown to simplify.

3.4 Statistical assessment

The MO-SAStrE optimization has been validated both graphically and in terms of hypervolume. However, these validations are not enough for the proposed approach, as it is necessary to know not only if alignments have been improved, but also if the improvement is statistically significant. To study that significance, the Kruskal–Wallis test was applied. The relevance of the 218 solutions was determined from the 10 runs per problem. This test provided *P*-values to confirm whether the output HV values were

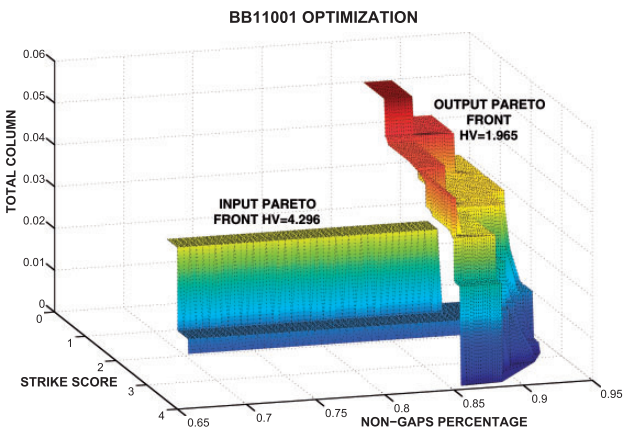

Fig. 5. 3D surfaces for the input and optimized Pareto fronts in the first problem of BAliBASE ('1aab' in Ref.1 v.1)

statistically different than the input ones. The significance level used to reject the null hypothesis in the 218 problems and to validate the improvement was set to $\alpha = 0.01$.

According to the proposed Kruskal–Wallis test, the complete dataset was considered significantly better, even those problems where improvements did not exceed 10%. Consequently, MO-SAStrE successfully optimized the 218 alignments with regard to the input methodologies because the null hypothesis was rejected in all of them. In addition, MO-SAStrE also returned better BAliscore results than the input alignments (Supplementary Table S3).

3.5 Comparison with other MSA methodologies

Finally, MO-SAStrE was compared with other genetic MSA methods, namely SAGA (Notredame and Higgins, 1996), MSA-GA (Gondro and Kinghorn, 2007), RBT-GA (Taheri and Zomaya, 2009) or DVGA (Naznin *et al*., 2011). Other non-genetic methodologies, which were also included in comparisons of previous genetic approaches, were also considered: ClustalW (Thompson *et al*., 1994), MultAlign (Barton and Sternberg, 1987), PRRP (Gotoh, 1996), PIMA (Smith and Smith, 1992), PILEUP (Devereux *et al*., 1984), Dialign (Morgenstern *et al*., 1996), HMMT (Eddy, 1995). These algorithms were assessed with a subset of problems in BAliBASE 2.0. However, as BAliBASE 3.0 was applied here, a subset of 20 problems included in both versions was selected. The 3D-COFFEE algorithm (O'Sullivan *et al*., 2004) was also added to compare MO-SAStrE against another important aligner using structural information.

Firstly, MO-SAStrE was compared with those methods included in MSA-GA and VDGA publications and 3D-COFFEE. MSA-GA defined two different configurations depending on whether a prealign procedure was included. Also, VDGA was configured according to the number of parts in which each sequence was decomposed: *Decomp_2*, *Decomp_3* or *Decomp_4*. Both tools were assessed against ClustalW. They included a set of 26 problems. MSA-GA and VDGA ran each of these problems five and ten times, respectively. The best result for each problem was then reported. Finally, the proposed solutions were evaluated with BAliscore. Here, a subset of eight problems, which were also included in BAliBASE v3.0, was considered to compare with MO-SAStrE. The BAliscore was also provided to measure the alignment accuracies. Additionally, 3D-COFFEE was also included to compare MO-SAStrE against another aligner using structural data. Both MO-SAStrE and 3D-COFFEE were run with all structures available in the PDB database for each specific set of sequences. Consequently, Table 3 shows the BAliscore results obtained from these methodologies (the best BAliscore values are highlighted in bold). From these eight problems, MO-SAStrE achieved a total of seven more accurate alignments against MSA-GA and VDGA, while it outperforms 3D-COFFEE in five out of the eight problems. Specifically, MSA-GA with the prealign procedure and VDGA Decomp_4 are better than MO-SAStrE in one case, mainly the '1uky' from Ref.1 v.1 subset. However, 3D-COFFEE achieves more accurate alignments in three Ref.1 v1 problems: '1ped', '1uky' and '2myr'. MO-SAStrE generally showed close BAliscore values to MSA-GA and VDGA but more distant than 3D-COFFEE in those problems where it does not achieve the most accurate alignment.

MO-SAStrE was also compared with SAGA, RBT-GA and, again, VDGA and 3D-COFFEE. Additionally, this comparison included other strategies used in the VDGA and RBT-GA assessment, namely PRRP, ClustalW, Dialign, PIMA, HMMT and PILEUP. Both RBT-GA and VDGA applied 10 independent runs for each problem and the best alignments were taken for their comparison. The subset proposed by RBT-GA and VDGA contained 34 problems from BAliBASE 2.0. Here, 12 of these problems were considered, as they must be included in both

Table 2. Average hypervolume

Subset	Input HV (Avg)	Output HV (Avg)	Improvement (%)
Ref.1 v.1	3.5894	0.7749	70.34
Ref.1 v.2	3.3004	0.7480	59.81
Ref.2	3.5478	0.6910	69.51
Ref.3	3.3003	0.5809	66.64
Ref.4	3.1347	0.7329	56.02
Ref.5	3.2268	0.8418	52.59

Note: HV values and improvements are shown according to the BAliBASE subsets.

Table 3. Comparison with MSA-GA, VDGA, ClustalW and 3D-COFFEE

Subset	Dataset name	MSA-GA	MSA-GA	CLUSTALW	VDGA (Decomp_2)	VDGA (Decomp_3)	VDGA (Decomp_4)	3DCOFFEE (v8.97)	MO-SAStrE
Ref.1 v.1	*1ped*	0.501	0.687	0.592	0.443	0.482	0.451	**0.812**	**0.716**
	1uky	0.443	0.405	0.392	0.416	0.459	**0.464**	**0.530**	0.403
	2myr	0.212	0.302	0.296	0.347	0.359	0.282	**0.675**	**0.544**
	Kinase	0.295	0.488	0.479	0.531	0.545	0.548	**0.783**	**0.808**
Ref.2	*1pamA*	0.755	0.758	0.757	0.857	0.863	0.853	**0.911**	**0.913**
	2pia	0.761	0.768	0.766	0.847	**0.850**	0.839	0.823	**0.879**
Ref.3	*Kinase*	0.580	0.619	0.619	0.870	0.890	0.887	**0.909**	**0.918**
Ref.4	*Kinase*	0.710	0.635	0.630	0.330	0.542	0.478	**0.863**	**0.865**

The BAliscore values are shown for 8 different BAliBASE datasets.
The two best scores are highlighted in bold.

Table 4. BAliscore comparison with SAGA, RBT-GA, VDGA and other known methodologies

Subset	Dataset name	PRRP	CLUSTALW	SAGA	DIALIGN	HMMT	PIMA (SB)	PIMA (ML)	MULT ALIGN	PILEUP8	RBT-GA	VDGA (Decomp_2)	VDGA (Decomp_3)	VDGA (Decomp_4)	3DCOFFEE (v8.97)	MO-SAStrE
Ref.2	*1lvl*	0.772	0.746	0.726	0.783	0.539	0.620	0.688	0.614	0.678	0.567	0.803	0.819	0.816	**0.827**	**0.825**
	1pamA	0.711	0.761	0.623	0.576	0.530	0.393	0.386	0.566	0.702	0.660	0.857	0.863	0.853	**0.911**	**0.913**
	1ubi	0.056	0.482	0.492	0.000	0.053	0.129	0.129	0.000	0.000	0.795	0.732	0.778	0.794	**0.901**	**0.911**
	1wit	0.760	0.557	0.694	0.724	0.641	0.469	0.463	0.500	0.476	0.825	0.875	0.815	0.774	**0.928**	**0.917**
	2hsdA	0.404	0.484	0.498	0.262	0.423	0.390	0.561	0.593	0.278	0.745	**0.856**	0.829	0.742	**0.888**	0.855
	2pia	0.767	0.752	0.763	0.612	0.647	0.730	0.695	0.765	0.766	0.730	0.847	**0.850**	0.839	0.823	**0.879**
	3grs	0.363	0.192	0.282	0.350	0.141	0.183	0.211	0.192	0.159	0.755	0.717	0.751	0.781	**0.861**	**0.864**
	4enl	0.668	0.375	0.739	0.122	0.213	0.096	0.092	0.384	0.224	0.812	0.890	0.889	0.899	**0.920**	**0.912**
Ref.3	*1ajsA*	0.128	0.163	0.186	0.000	0.006	0.000	0.000	0.000	0.110	0.180	0.383	0.453	0.408	**0.572**	**0.586**
	1ubi	0.415	0.146	**0.585**	0.000	0.366	0.000	0.000	0.000	0.268	0.310	0.398	0.414	0.41	0.525	**0.590**
	1uky	0.139	0.130	0.269	0.139	0.037	0.083	0.148	0.241	0.083	0.350	0.469	0.481	0.526	**0.625**	**0.673**
	4enl	0.736	0.547	0.672	0.050	0.050	0.393	0.438	0.652	0.498	0.680	0.836	**0.866**	**0.866**	0.853	0.862

The BAliscore values are shown for 12 BAliBASE datasets.
The two best scores are highlighted in bold.

Table 5. Wilcoxon non-parametric test

MSA tool	Sign+	Sign−	Z	*P*-value	$P<0.05$
MSA-GA	7	1	−2.381	0.017	Yes
MSA-GA prealign	7	1	−2.381	0.017	Yes
PRRP	12	0	−3.059	0.002	Yes
SAGA	12	0	−3.059	0.002	Yes
DIALIGN	12	0	−3.059	0.002	Yes
HMMT	12	0	−3.059	0.002	Yes
SB_PIMA	12	0	−3.059	0.002	Yes
ML_PIMA	12	0	−3.059	0.002	Yes
MULTALIGN	12	0	−3.059	0.002	Yes
PILEUP8	12	0	−3.059	0.002	Yes
RBT-GA	12	0	−3.059	0.002	Yes
CLUSTALW	20	0	−3.920	0.000	Yes
VDGA_Decomp2	18	2	−3.809	0.000	Yes
VDGA_Decomp3	18	2	−3.510	0.000	Yes
VDGA_Decomp4	18	2	−3.547	0.000	Yes
3D-COFFEE	13	7	−0.579	0.562	No

Note: Pairwise comparisons between MO-SAStrE and each other method. 'Sign+'/'Sign−' identifies the number of problems that MO-SAStrE won/lost the other method, respectively. 'Z' is the score provided by the Wilcoxon test.

versions of BAliBASE. The obtained BAliscore results are presented in Table 4. From this table, it is observed that MO-SAStrE achieves one of the two best results in 10 out of 12 problems. VDGA outperforms MO-SAStrE in three problems distributed into its three decompositions, while 3D-COFFEE achieves better alignments in four problems. In this second comparison, the alignments where MO-SAStrE achieves worse results are closer to the best accuracy than those alignments proposed by other methodologies (excepting 3D-COFFEE whose alignments are similar).

Finally, both comparisons (Tables 3 and 4) were joined to estimate the significance of the MO-SAStrE improvement. Specifically, the BAliscore results were compared with the Wilcoxon non-parametric statistical test (Table 5). According to the obtained *P*-values, MO-SAStrE shows significant improvements over the other methods, except 3D-COFFEE. This improvement is even more relevant comparing with ClustalW and VDGA, as a larger number of problems (20 alignments) were used. Regarding the comparison with 3D-COFFEE, both 3D-COFFEE and MO-SAStrE provide similar alignments, as none achieves a statistically significant improvement against the other (Wilcoxon test, $P>0.05$). Nevertheless, MO-SAStrE includes some additional advantages with respect to 3D-COFFEE. Firstly, MO-SAStrE is able to work with only one structure, whereas 3D-COFFEE requires at least two structures to build the structure superposition. Therefore, the alignment accuracy from 3D-COFFEE directly depends on the number of available structures, making it less useful when just a few structures are available. Additionally, MO-SAStrE provides more flexibility owing to the fact that it includes other optimization criteria (multiobjective approach) to evaluate and improve the alignments in addition to structural information.

3.6 PDB structure availability

We acknowledge that one of the main drawbacks of the proposed approach could be the limited availability of PDB structures. However, the pivotal relevance of structures to accomplish well-annotated sequences is beyond dispute. Thus, PDB is currently making a major effort to accurately annotate proteins' structures, which has been translated into an exponential increase in the past 10 years (87 089 structures in 2012). It is also known that current databases have admitted this relevance and they are currently being updated to include as many PDB structures as possible. For instance, the Pfam database (Punta *et al.*, 2012), which identifies a number of families related by common functional domains, considers that the use of structural information will help to improve domain definitions and to increase coverage of sequences included in other databases. Thus, Pfam database (release 24.0) (Punta *et al.*, 2012) already includes some structural annotation in almost 50% of its families, which represents the 95% of the known PDB structures.

Additionally, it is important to highlight that the main goal of MSA tools is to compare unknown sequences with those well-annotated ones to infer several biological features of such

sequences. Then, at least one well-annotated sequence should be included in MSAs, including at least some structural information. Anyway, to make the proposed approach more robust, MO-SAStrE implements an alternative objective for those cases where sequences lack any PDB structures. In those cases, the STRIKE objective is substituted by an easier evaluation such as the PAM250 score (Dayhoff *et al*., 1978). Although this alternative is not the main goal here, it was also checked that the multiobjective optimization using the PAM250 score could also be effective.

4 CONCLUSIONS

Currently, MSAs are an open issue for researchers. Aligners must be continually improved, as they are essential in the analysis of huge amount of data provided by next-generation sequencing and high-throughput experiments. For this reason, the most efficient computational techniques are fundamental to reduce the cost of analyzing new information and to improve the obtained accuracy.

A complete algorithm called MO-SAStrE was proposed to optimize MSAs. This algorithm was developed through the multiobjective approach NSGA-II, specially based on a structure evaluation (STRIKE score). Then, this algorithm takes advantage of a wider range of optimization measures than other similar methodologies. Although this is not the main purpose, the PAM250 score could also be applied for the first objective in case of sequences not having structures available. For this algorithm, alignments previously obtained from eight methodologies (mainly progressive and consistency-based ones) were coded using a novel representation and own-designed crossover/mutation procedures. The obtained alignments were built as an ensemble of the best aligned parts from these solutions to adjust the sequences as precisely as possible. The results for this approach showed that the alignments could generally be improved without the application of more time-consuming aligners. A complete set of problems from BAliBASE 3.0 was then applied. The HV and the Kruskal–Wallis test confirmed that MO-SAStrE achieves significantly optimized alignments with regard to the input methodologies. Additionally, comparisons with other genetic and non-genetic approaches showed that MO-SAStrE can provide more accurate alignments according to the BAliscore results.

Funding: Spanish CICYT Project [SAF2010-20558 (in part)] and the government of Andalusia Project [P09-TIC-175476].

Conflict of Interest: none declared.

REFERENCES

Altschul,S. *et al.* (1997) Gapped blast and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res.*, **25**, 3389–3402.

Bacon,D.J. and Anderson,W.F. (1986) Multiple sequence alignment. *J. Mol. Biol.*, **191**, 153–161.

Barton,G.J. and Sternberg,M.J.E. (1987) A strategy for the rapid multiple alignment of protein sequences—confidence levels from tertiary structure comparisons. *J. Mol. Biol.*, **198**, 327–337.

Berman,H. *et al.* (2000) The protein data bank. *Nucleic Acids Res.*, **28**, 235–242.

Bradley,R.K. *et al.* (2009) Fast statistical alignment. *PLoS Comput. Biol.*, **5**, e1000392.

Chou,P. and Fasman,G. (1978) Prediction of the secondary structure of proteins from their amino acid sequence. *Adv. Enzymol. Relat. Areas Mol. Biol.*, **47**, 145–148.

Conover,W.J. (1999) *Practical Nonparametric Statistics*. 3rd edn. Wiley, New York.

Dasgupta,D. *et al.* (2009) On the use of informed initialization and extreme solutions sub-population in multiobjective evolutionary algorithms. In: *MCDM: 2009 IEEE Symposium on Computational Intelligence in Multi-criteria Decision-Making*. pp. 58–65.

Dayhoff,M.O. *et al.* (1978) A model of evolutionary change in proteins. In: Dayhoff,M.O. (ed.) *Atlas of Protein Sequence and Structure*. National Biomedical Research Foundation. Vol. 5. Washington, DC, pp. 345–352.

Deb,K. *et al.* (2002) A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Trans. Evolut. Comput.*, **6**, 182–197.

Devereux,J. *et al.* (1984) A comprehensive set of sequence-analysis programs for the vax. *Nucleic Acids Res.*, **12**, 387–395.

Do,C. *et al.* (2005) ProbCons: probabilistic consistency-based multiple sequence alignment. *Genome Res.*, **15**, 330–340.

Doolittle,R.F. (1981) Similar amino acid sequences: chance or common ancestry? *Science*, **214**, 149–159.

Eddy,S.R. (1995) Multiple alignment using hidden Markov models. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **3**, 114–120.

Edgar,R. (2004) MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res.*, **32**, 1792–1797.

Eiben,A.E. and Smith,J.E. (2008) *Introduction to evolutionary computing. (Natural Computing Series)*. Springer, Berlin, Germany.

Feng,D. and Doolittle,R. (1987) Progressive sequence alignment as a prerequisite correct phylogenetic trees. *J. Mol. Evolut.*, **25**, 351–360.

Fitch,W.M. (1966) An improved method of testing for evolutionary homology. *J. Mol. Biol.*, **16**, 9–16.

Gondro,C. and Kinghorn,B.P. (2007) A simple genetic algorithm for multiple sequence alignment. *Genet. Mol. Res.*, **6**, 964–982.

Gotoh,O. (1990) Consistency of optimal sequence alignments. *Bull. Math. Biol.*, **52**, 509–525.

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.

Henikoff,S. and Henikoff,J.G. (1992) Amino-acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. USA*, **89**, 10915–10919.

Hogeweg,P. and Hesper,B. (1984) The alignment of sets of sequences and the construction of phyletic trees: an integrated method. *J. Mol. Evolut.*, **20**, 175–186.

Katoh,K. *et al.* (2002) MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform. *Nucleic Acids Res.*, **30**, 3059–3066.

Kececioglu,J. *et al.* (2010) Aligning protein sequences with predicted secondary structure. *J. Comput. Biol.*, **17**, 561–580.

Kemena,C. and Notredame,C. (2009) Upcoming challenges for multiple sequence alignment methods in the high-throughput era. *Bioinformatics*, **25**, 2455–2465.

Kemena,C. *et al.* (2011) STRIKE: evaluation of protein msas using a single 3d structure. *Bioinformatics*, **27**, 3385–3391.

Kruskal,W.H. and Wallis,W.A. (1952) Use of ranks in one-criterion variance analysis. *J. Am. Stat. Assoc.*, **47**, 583–621.

Lassmann,T. and Sonnhammer,E. (2005) Kalign—an accurate and fast multiple sequence alignment algorithm. *BMC Bioinformatics*, **6**, 298.

Li,H. and Homer,N. (2010) A survey of sequence alignment algorithms for next-generation sequencing. *Brief. Bioinform.*, **11**, 473–483.

Lin,K. *et al.* (2003) Testing homology with contact accepted mutation (CAO): a contact-based Markov model of protein evolution. *Comput. Biol. Chem.*, **27**, 93–102.

Liu,K. *et al.* (2009) Rapid and accurate large-scale coestimation of sequence alignments and phylogenetic trees. *Science*, **324**, 1561–1564.

Mirarab,S. and Warnow,T. (2011) Fastsp: linear time calculation of alignment accuracy. *Bioinformatics*, **27**, 3250–3258.

Morgenstern,B. *et al.* (1996) Multiple dna and protein sequence alignment based on segment-to-segment comparison. *Proc. Natl. Acad. Sci. USA*, **93**, 12098–12103.

Naznin,F. *et al.* (2011) Vertical decomposition with genetic algorithm for multiple sequence alignment. *BMC Bioinformatics*, **12**, 353.

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

Notredame,C. *et al.* (2000) T-Coffee: a novel method for fast and accurate multiple sequence alignment. *J. Mol. Biol.*, **302**, 205–217.

Nozaki,Y. and Bellgard,M. (2005) Statistical evaluation and comparison of a pairwise alignment algorithm that a priori assigns the number of gaps rather than employing gap penalties. *Bioinformatics*, **21**, 1421–1428.

Nuin,P.A. *et al.* (2006) The accuracy of several multiple sequence alignment programs for proteins. *BMC Bioinformatics*, **7**, 471.
O'Sullivan,O. *et al.* (2004) 3DCoffee: combining protein sequences and structures within multiple sequence alignments. *J. Mol. Biol.*, **340**, 385–395.
Pei,J. and Grishin,N.V. (2007) PROMALS: towards accurate multiple sequence alignments of distantly related proteins. *Bioinformatics*, **23**, 802–808.
Punta,M. *et al.* (2012) The Pfam protein families database. *Nucleic Acids Res.*, **40**, D290–D301.
Redelings,B. and Suchard,M. (2005) Joint bayesian estimation of alignment and phylogeny. *Syst. Biol.*, **54**, 401–418.
Ronquist,F. and Huelsenbeck,J. (2003) Mrbayes 3: bayesian phylogenetic inference under mixed models. *Bioinformatics*, **19**, 1572–1574.
Schneider,T.D. *et al.* (1986) Information content of binding sites on nucleotide sequences. *J. Mol. Biol.*, **188**, 415–431.
Sierk,M.L. *et al.* (2010) Improving pairwise sequence alignment accuracy using near-optimal protein sequence alignments. *BMC Bioinformatics*, **11**, 146.
Smith,R.F. and Smith,T.F. (1992) Pattern-induced multi-sequence alignment (pima) algorithm employing secondary structure-dependent gap penalties for use in comparative protein modeling. *Protein Eng.*, **5**, 35–41.
Szabo,A. *et al.* (2010) Reticular alignment: a progressive corner-cutting method for multiple sequence alignment. *BMC Bioinformatics*, **11**, 570.
Taheri,J. and Zomaya,A.Y. (2009) RBT-GA: a novel metaheuristic for solving the multiple sequence alignment problem. *BMC Genomics*, **10(Suppl 1)**, S10.
Taylor,W.R. and Thornton,J.M. (1984) Recognition of super-secondary structure in proteins. *J. Mol. Biol.*, **173**, 487–514.
Thompson,J. *et al.* (1994) ClustalW: improving the sensivity of progressive multiple sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic Acids Res.*, **22**, 4673–4680.
Thompson,J. *et al.* (2005) BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins*, **61**, 127–136.
Tsujimoto,Y. *et al.* (2009) Effects of including single-objective optimal solutions in an initial population on evolutionary multiobjective optimization. In: *2009 International Conference of Soft Computing and Pattern Recognition*. pp. 352–357.
Wilcoxon,F. (1945) Individual comparisons by ranking methods. *Biometr. Bull.*, **1**, 80–83.
Wong,K.M. *et al.* (2008) Alignment uncertainty and genomic analysis. *Science*, **319**, 473–476.
Zitzler,E. *et al.* (2008) Quality assessment of pareto set approximations. In: Branke,J., Deb,K., Miettinen,K. and Slowiński,R. (eds) *Multiobjective Optimization: Interactive and Evolutionary Approaches*. Vol. 5252, Springer, Berlin, Germany, pp. 373–404.