Protein Progressive MSA Using 2-Opt Method

Gamil Abdel-Azim[1,2], Aboubekeur Hamdi-Cherif[1,3]
Mohamed Ben Othman[1,4] and Z.A. Aboeleneen[5]
[1]*College of Computer, Qassim University*
[2]*College of Computer& Informatics, Canal Suez University*
[3]*Computer Science Department, Université Ferhat Abbas, Setif (UFAS)*
[4]*The research Unit of Technologies of Information and Communication (UTIC) / ESSTT*
[5]*College of Computer& Informatics, Zagazig University*
[1]*Saudi Arabia*
[3]*Algeria*
[4]*Tunisia*
[2,5]*Egypt*

1. Introduction

Multiple sequence alignment (MSA) is a very useful tool in designing experiments for testing and modifying the function of specific proteins, in predicting their functions and structures, and in identifying new members of protein families. MSA of deoxyribonucleic acid (DNA), ribonucleic acid (RNA) and protein remains one of the most common and important tasks in Bioinformatics. Textbooks on the algorithms dedicated to sequence alignment appeared more than a decade ago, e.g. (Durbin et al., 1998). Many critical overviews of the existing MSA have been investigated (Notredame, 2002; Kumar & Filipski, 2007). Finding an optimal MSA of a given set of sequences has been identified as a nondeterministic polynomial-time (NP)-complete problem (Wang & Jiang, 1994). The MSA solution, based on dynamic programming, requires $O((2m)^n)$ time complexity; n being the number of sequences, and m the average sequence length. The memory complexity is $O(m^n)$ (Carrillo & Lipman, 1988; Saitou & Nei, 1987). Therefore, carrying out MSA by dynamic programming becomes practically intractable as the number of sequences increases. The dynamic programming algorithm used for optimal alignment of pairs of sequences can easily be extended to global alignment of three sequences. But for more than three sequences, only a small number of relatively short sequences may be analyzed because of the "curse of dimensionality". Despite the existence of many ready-made and operational systems such as *MBEToolbox* (Cai et al., 2006), *Probalign* (Roshan & Livesay, 2006), *Mulan* (Loots & Ovcharenko, 2007), MSA is always an active area of research (Yue et al., 2009). Approximate methods are constantly investigated for global MSA. One class of these methods is the progressive global alignment. The method starts with an alignment of the most-alike sequences and then builds an alignment by adding more and more closely-alike sequences. Progressive alignment was first formulated in (Hogeweg & Hesper, 1984). Progressive alignment, as implemented in some packages such as ClustalW, for instance, represents one the most popular methodology for MSA. However, in ClustalW, alignment is made by the explicit use of the sequences themselves, which certainly represents a heavy computational burden (Thompson et al., 1994). Building on previous works such as (Azim et al., 2010; 2011), and in order to reduce this computational effort, we represent the similarity between the sequences using a descriptor of the sequences instead of the sequences themselves. The main advantage of using the proposed descriptor resides in its short length, namely 20 for proteins and 4 for DNA, irrespective of the sequence length. Based on this idea, a novel descriptor-based progressive MSA, called DescPA, is formulated, and further improved through a 2-opt method resulting in DescPA2. This novel approach positively impacts the computation time for the MSA, as shown in the results. The chapter is organized as follows. In the next section, the description of protein MSA problem is highlighted. Section 3 briefly presents the DescPA steps as a novel methodology using the Hellinger distance and the computation of the probability density functions (PDF) of sequences. Section 4 reports further enhancements through DescPA2 based on a local search method, namely 2-opt. Section 5 reports the results for both DescPA and DescPA2 performance with respect to ClustalW. Finally, concluding remarks and further research are presented.

2. Proteins MSA problem formulation

2.1 MSA at large

2.1.1 The MSA difficulties

MSA is an interdisciplinary problem. It spans three distinct fields, namely statistics, biology and computer science; each of which encompassing technical difficulties, summarized in the choices of :

- the sequences,
- an objective function (i.e., a comparison model),
- the optimization method for that function.

As a result, properly solving these three problems would require an understanding of all three fields mentioned above, which obviously lies far beyond our reach.

2.1.2 Sequence choice issues

The global of MSA methods assume that we are dealing with a set of homologous sequences i.e., sequences sharing a common ancestor. Furthermore, with the exception of some methods (e.g. Morgenstern et al., 1996), MSA solutions require the sequences to be related over their whole length (or at least most of it). When that condition is not met, one has to rely on the use of local MSA methods such as a sampler (Lawrence et al., 1993), among others.

2.1.3 Objective or cost function issue

The objective or cost function is the mathematical formulation of a purely biological problem that lies in the definition of biological correctness. A mathematical function is used for measuring the biological quality of an alignment. This function is referred to as an objective or cost function since it defines the mathematical objective or cost of the search. Given a perfect function, the mathematically optimal alignment will also be biologically optimal. Unfortunately, this is rarely the case, and while the function defines a mathematical optimum, we rarely have a sound argument that this optimum will also be biologically optimal. As a result, an ideal objective or cost function for all situations does not exist, and every available scheme suffers from major drawbacks. Ideally, a perfect objective or cost function is to be available for every situation. In practice, this is not the case and the user is always left to make a decision when choosing the method that is most suitable to the problem (Durbin et al., 1998).

2.1.4 Optimization issue

The third main issue associated with MSAs is purely computational. If we assume that we have an adequate set of sequences and a biologically perfect objective function, there still remain the optimization of the objective or cost function. This task is far from being trivial. the computation of a mathematically optimal alignment is too complex a task for an exact method to be used (Wang & Jiang, 1994). Even if we consider a function that consists of the maximization of the number of perfect identities within each column, the problem would still remain intractable for more than three sequences. Consequently, all the current implementations of multiple alignment algorithms are heuristics and that none of them guarantee a full optimization.

2.2 Existing MSA optimization algorithms

Algorithms that construct MSA require a cost function as a criterion for constructing an optimal alignment. There exist three categories of MSA optimization; exact, iterative and progressive (Saitou & Nei, 1987). The exact method suffers from inexact sequence alignment (Wang & Li, 2004). Commonly-used techniques remain the iterative and progressive techniques. Most progressive MSA methods heavily rely on dynamic programming to perform multiple alignments starting with the most closely-related sequences and then progressively adding other related sequences to the initial alignment. These methods have the advantage of being fast, simple as well as reasonably sensitive. Their main drawback is that they can be trapped in local minima that stems from the greedy nature of the algorithm (Thompson et al. 2005). The other major drawback is that any progressive MSA solution cannot be globally optimal, since it is heavily influenced by the initial choice. As a result, any error made at any stage in building the MSA, is propagated and builds up through to the final result. Finally, the performance gets worse when all the sequences in the set are rather distantly-related. Despite these limitations, progressive alignment methods are still efficient enough to be implemented on a large scale for hundreds to thousands of sequences. Hence our contribution to their enhancement.

2.3 Progressive strategy

2.3.1 Basic steps

The existence of several progressive programs and packages has broadened up the aligning techniques. The most popular progressive MSA implementation is represented in the ClustalW family (Higgins & Sharp, 1988; Thompson et al., 1994; 2005). The guide tree in the basic progressive strategy is determined by an efficient clustering method such as neighbor-joining (Saitou & Nei, 1987), or un-weighted average distance (Carrillo & Lipman, 1988).

The progressive strategy, also known as tree method, is one of the most widely used heuristic search for MSA. It combines pairwise alignments beginning with the most similar pair and progressing to the most distantly-related, which finally builds up an MSA solution. The basic progressive alignment strategy follows three steps, depicted in Fig. 1, below.

```
// Basic progressive strategy //

1. Compute D, a matrix of distances between all pairs of sequences.
2. From D, construct a guide tree T.
3. Process the progressive alignment: construct MSA by pairwise alignment
   of partial alignments (profiles) guided by T.
```

Fig. 1. Basic progressive strategy

2.3.2 Introductory example

Let $S = \{S_1, S_2, \ldots, S_n\}$ be the input sequences and assume that n is at least 2. Let Σ be the input alphabet that form the sequences. We assume that Σ does not contain the gap character '–'. Any set $S' = \{S'_1 , S'_2 , \ldots, S'_n \}$ of sequences over the alphabet $\Sigma' = \Sigma \cup \{–\}$, is called an alignment of S if the following two properties satisfied :

1. 1. The strings in S' have the same length.
2. 2. Ignoring gaps, sequences S_i' and S_i are identical.

An alignment can be interpreted as a matrix with n rows and m columns; one row for each S_i and one column for each character in Σ'. Two letters of distinct strings are said to be aligned under S if they are placed into the same column.

For example, Figure 2 shows an alignment for three proteins sequences.

$$AS = \begin{bmatrix} A & R & N & - & D & C & Q & E & G & H & I & L & M & F & - & W & T & W & Y & V \\ - & R & - & N & D & C & Q & E & G & H & I & L & M & F & S & - & T & W & Y & V \\ A & R & N & - & D & C & Q & E & G & H & I & L & M & F & S & - & T & W & Y & V \end{bmatrix}$$

Fig. 2. MSA introductory example for three proteins sequences

3. Descriptor-based progressive MSA (DescPA)

3.1 Basic DescPA

3.1.1 Outline

Within the Clustal-like family, we propose a novel measurement method of the similarity between the sequences, which plays an important role in the building of the guide tree. This measurement is based on the calculation of the probability density function (PDF), also called descriptor or feature vector sequence. The descriptor reduces the dimension of the sequence and yields to a faster calculation of the distance matrix and also to the obtainment of a preliminary distance matrix without pairwise alignment in the first step. For achieving this goal, we use a guide tree based on Hellinger distance. This latter is defined between the descriptors and measures the degree of similarity between the sequences.

3.1.2 DescPA steps

We briefly describe the basic steps of the proposed method, referred to as the descriptor-based progressive MSA (DescPA), outlined in Fig. 3, below.

```
// DescPA Steps //

1. Read set of proteins sequences.
2. Estimate the distance matrix between all sequences.
3. Construct the guide tree using distance matrix methods.
4. Apply the progressive alignment methods with guide tree.
5. Output the resulting sequences alignments.
```

Fig. 3. Steps for DescPA

3.1.3 Overall architecture of DescPA

As shown in Figures 3 and 4, the proposed algorithm consists of 3 phases similar to ClustalW. The main difference with ClustalW resides in the way in which the distance matrix is built, here based on Hellinger distance. Each sequence descriptor is described by its probability density function (PDF). The guide tree defines the order in which the sequences are aligned in the next stage. There are several methods for building trees, including distance matrix methods and parsimony methods.

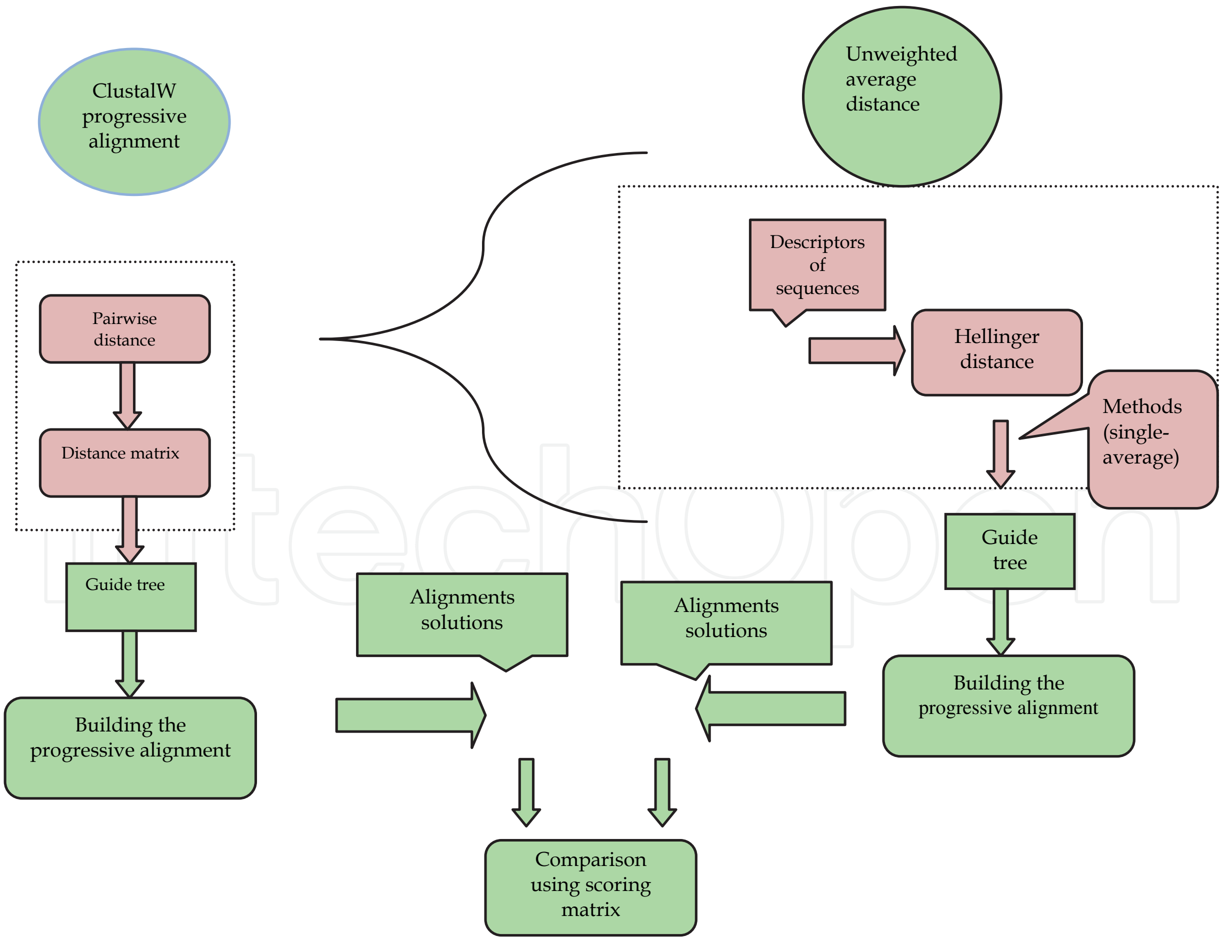


Fig. 4. Overall architecture of DescPA as compared with ClustalW

3.2 Mathematical tools

We need to define some of the basic mathematical tools, necessary for the development of our method. These methods include the Hellinger distance, the PDF calculation, and the scoring matrices.

3.2.1 Hellinger distance

The Hellinger distance is a metric quantity, meaning that it has the properties of non-negativity, identity and symmetry in addition to obeying the triangle inequality. The properties of the Hellinger distance and several related distances are explored in (Donoho & Liu, 1988; Giet & Lubrano, 2008). This concept is used to provide a metric for the distance between two different discrete probability distributions *P* and *Q*, as follows:

$$D^2(P,Q)=\frac{1}{2}\sum_{i=1}^{N}(\sqrt{p_i}-\sqrt{q_i})^2 \qquad (1)$$

Note that P and Q are described as *N*-tuples (vectors) of probabilities $(p_{1,}p_2,...,p_N)$ and $(q_{1,}q_2,...,q_N)$ where p_i and q_i are assumed to be non-negative real numbers with:

$$\sum_{i=1}^{N} p_i = 1; \qquad \sum_{i=1}^{N} q_i = 1 \qquad (2)$$

3.2.2 Computing the probability density functions (PDFs)

We can compute the Hellinger distance between two variables provided we have explicit knowledge of the probability distributions. Unfortunately, these probabilities are not known in general. Various methods are used to estimate the probability density functions (PDFs) from the observed data. In this paper, we calculate exact probability densities for each proteins sequence. Consider a series x_i and y_i of *n* simultaneous observations of two random variables *X* and *Y*. Since Hellinger distance is computed using discrete probabilities, we proceed as follows:

Let $f_X(i)$ denotes the number of observations i in X . The probabilities p_i are then estimated as:

$$p_i = \frac{f_X(i)}{n} \qquad (3)$$

Similarly, let $f_Y(j)$ denote the number of observations of j in *Y*. The probabilities q_j are then estimated as:

$$q_j = \frac{f_Y(j)}{n} \qquad (4)$$

Then the Hellinger distance between *X* and *Y* is estimated using Equation (1) above

The descriptor is defined as follows:

$$f : prot \rightarrow [0,1]^n \qquad (5)$$

Where *prot* is the set of proteins sequences. The proteins alphabet is given by the 20-character set { A R N D C Q E G H I L K M F P S T W Y V }. The descriptor is calculated for each protein sequence as the PDF of the sequence, obtained as follows:

$$p_i = \frac{N_i}{len(S_i)} \tag{6}$$

where:
$len(S_i)$ is the length of the sequence,
N_i is the number of times character *i* appears in the sequence.
i belongs to the proteins 20-character alphabet.

3.2.3 Scoring matrices

(i) PAM vs. BLOSUM

Various scoring matrices exist. The main ones are the so-called PAM and BLOSUM (Wheeler, 2003). The most widely used PAM matrix is PAM 250. It has been chosen because it is capable of accurately detecting similarities in the 30% range, that is, when the two proteins are up to 70% different from each other. If the goal is to know the widest possible range of proteins similar to the protein of interest, PAM 250 has been shown to be the most effective. It is also the best to use when the protein is unknown or may be a fragment of a larger protein. Based on an information-theoretic measure called relative entropy it has been shown that the following matrices are equivalent (Henikoff and Henikoff, 1992):

- PAM 250 is equivalent to BLOSUM45.
- PAM 160 is equivalent to BLOSUM62.
- PAM 120 is equivalent to BLOSUM80.

Recall that PAM matrices are the result of computing the probability of one substitution per 100 amino acids, called the PAM1 matrix. Higher PAM matrices are derived by multiplying the PAM1 matrix by itself a defined number of times. Thus, the PAM250 matrix is derived by multiplying the PAM1 matrix against itself 250 times. Biologically, the PAM250 matrix means there have been 2.5 amino acid replacements at each site (Wheeler, 2003).

In the derivation of PAM matrices, sequences that were represented many times were not excluded from the calculation. During the construction of BLOSUM (Blocks Substitution Matrix) matrices, measures were taken to avoid biasing the matrices by removing frequently occurring and highly related sequences. Consequently, as the BLOSUM number decreases (i.e., BLOSUM80, BLOSUM60, BLOSUM50, BLOSUM30...), the ability to detect more distantly related sequences increases in a manner that parallels the effect of increasing the PAM distance (i.e., PAM 40, PAM160, PAM250...), (Altschul, 1991).

(ii) Gonet matrix

In addition to PAM250, we used Gonnet matrix. The Gonnet matrix is a scoring matrix based on alignment of the entire 1991 SwissProt database against itself (Gonnet et al., 1992). A total of 1.7 × 106 matches were used from sequences differing by 6.4 to 100.0 PAM units. This matrix has broad but selective coverage of protein sequences, because SwissProt covers only selected families. This matrix is very useful because of the excellent annotation of proteins included in SwissProt (Wheeler, 2003).

3.2.4 Summarized calculations sub-steps

Fig. 5 below describes the calculations sub-steps undertaken by DescPA.

```
// DescPA calculations sub-steps //

1. Define the descriptor as PDF.
2. Find the PDF for each of two sequences $S_i$ and $S_j$.
3. Calculate the Hellinger distance between $S_i$ and $S_j$ using Equation (1)
```

Fig. 5. Calculations sub-steps for DescPA

4. Hybridization with 2-opt method

4.1 Local search as improvement methodology

About 93% of the results obtained with basic DescPA compare well with those of ClustalW, as shown in Section 5.1 below, but they are not better. This motivates for the introduction of an enhancement method. A local search method is a good candidate for such an improvement. The resulting improved implementation is referred to as DescPA2. Iterative local search methods rely on algorithms that are able to produce a solution and to refine it, through a series of iterations until no improvement can be made (Wang & Li, 2004), *e.g.* genetic algorithms as local optimizers (Wang & Lefkowitz, 2005). In our study we propose a local search, that starts from initial solution (*i.e.* alignment) and repeatedly tries to improve the current solution by local change. If, in the neighborhood of the current alignment a better alignment is found, then it replaces the current one and local search continues. The critical issue in the design of a neighborhood search approach is the choice of the neighborhood structure. In this work, the neighborhood of a solution is depends on the neighborhood $N(\rho)$ of the permutation ρ that is defined by the set of all possible permutations, obtained by exchanging 2 elements. The neighborhood structure $N(PS)$ of the solution is defined as:

$$N(PS) = \bigcup_{(i=1,2,\ldots,n)} N(\rho_i) \tag{6}$$

4.2 The 2-opt method

4.2.1 Outline of the method

The 2-opt method is a combinatorial optimization method originating in the late 1950's in conjunction with the traveling salesman problem (Johnson & McGeoch, 1997). As an adaptation, we define the permutation solution's space corresponding to alignment solution's space. We define the function $\varphi(S,\rho) = S_\rho$ for each sequence S and permutation ρ as follows:

$$S_{\rho(i)} = \begin{cases} S_{\rho'(i)} & i = 1,2,\ldots,l \\ - & i = l+1,\ldots,m \end{cases} \tag{7}$$

where $\rho^{'}$ is the first sorted elements (sub-permutation) of ρ,

m is the dimension of ρ and l is length of the sequence S. Then by using the definition 7, we can associate permutation solution PS for each alignment solution AS.

4.2.2 Example

Figure 6 illustrates the use of definition 7 with permutation solution *PS*.

Sequence	Length	Permutation	Sub permutation	Sorted permutation
ATCAA CGTAGTG TGATCT	5 7 6	(3 5 8 9 1 7 2 4 6) (6 7 4 9 1 3 5 2 8) (7 6 3 2 1 5 8 9 4)	(3 5 8 9 1) (6 7 4 9 1 3 5) (7 6 3 2 1 5)	(1 3 5 8 9) (1 3 4 5 6 7 9) (1 2 3 5 6 7)
Alignment solution			Permutation solution (structure)	
$AS = \begin{bmatrix} \text{A - T - C - - A A} \\ \text{C - G T A G T - G} \\ \text{T G A - T C T - -} \end{bmatrix}$			$PS = \begin{bmatrix} (3\,5\,8\,9\,1\,7\,2\,4\,6) \\ (6\,7\,4\;\,9\,1\,3\,5\,2\,8) \\ (7\,6\,3\;\,2\,1\,5\,8\,9\,4) \end{bmatrix}$	

Fig. 6. Illustration of the definition of permutation using 3 sequences.

5. Results

5.1 ClustalW *vs.* DescPA results

We compare DescPA with ClustalW using 2 examples. Here, 4 and 9 proteins sequences are used with minimum lengths of 390, and 385 and maximum lengths of 456 and 457, respectively. For both examples, a comparison is made between the results obtained using pairwise (ClustalW) and Hellinger distances (DescPA). We implement the two guide trees using Matlab™ functions as described below.

5.1.1 Guide trees construction

1. **`TreePW = seqlinkage(DistancePW,'single',seqs)`**, where **`seqlinkage`** is a Matlab™ function, that implements neighbor-joining algorithm.
2. **`DistancePW = seqpdist(seqs,'ScoringMatrix', pam250)`**, where **`seqs`** are the proteins sequences.
3. **`TreeHD = seqlinkage (HD,'single',seqs)`**, where **`HD`** is the proposed Hellinger distance matrix.

Figures 7&8 give the comparison between ClustalW (**`TreePW)`** and DescPA (**`TreeHD)`** with solution alignment scoring values of the 2 proposed examples over the datasets of BAliBASE 3.0 (Thompson, 2005).

5.1.2 Data set used

The information concerning the data set taken from the database is summarized as follows (Bahr et al., 2001).
Reference 1: Equidistant sequences with 2 different levels of conservation.
Reference 2: Families aligned with a highly divergent "orphan" sequence.
RV11: Reference 1, very divergent sequences (20 identity)
RV12: Reference 1, medium-divergent sequences (20-40 identity).
RV20: Reference 2.
The progressive algorithm is implemented as a Matlab™ function (Version 7.0) called **`multialign`** which can be used with the following options:

`multialign` (**`S`**, **`'terminalGapAdjust',true`**).

(i) Example 1: Aligning 4 proteins

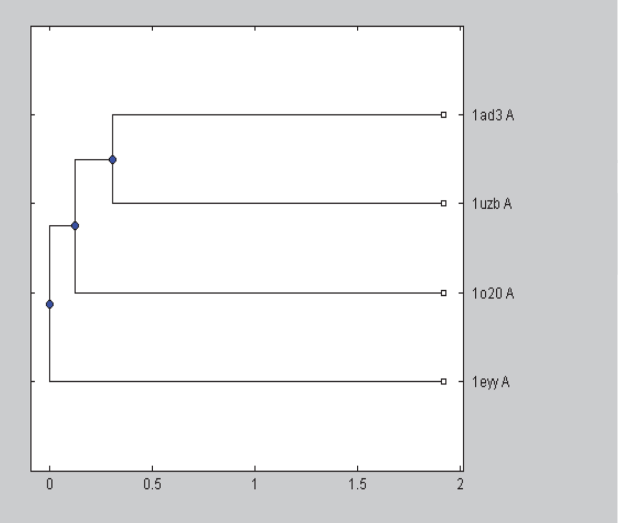
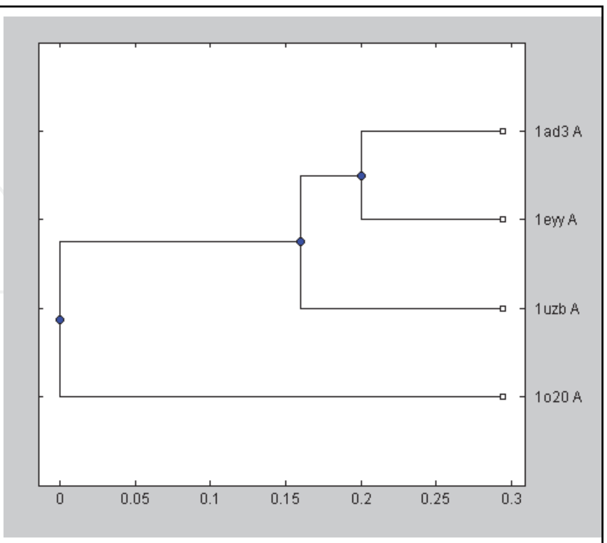
	
ClustalW **`(TreePW)`** with pairwise distance Scoring value is 144.7000	DescPA **`(TreeHD)`** with proposed Hellinger distance Scoring value is 148.4000

Fig. 7. Tree of solutions for ClustalW (**`TreePW`**) and DescPA (**`TreeHD)`** for 4 proteins

(ii) Example 2: Aligning 9 proteins

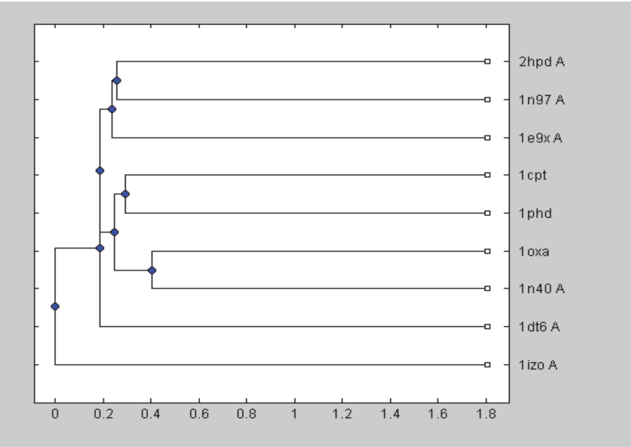
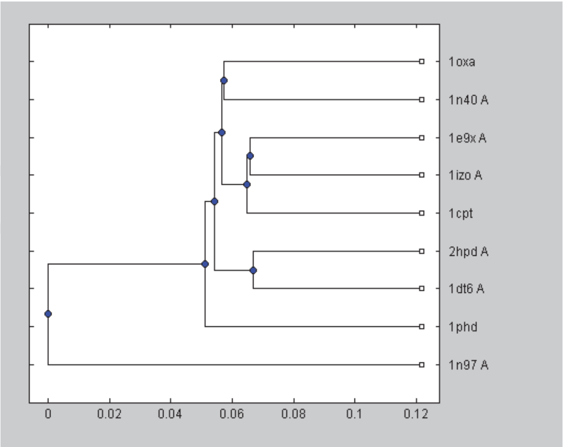
	
ClustalW **`(TreePW)`** with pairwise distance. Scoring value = 2.2277e+003	DescPA **`(TreeHD)`** with proposed Hellinger distance Scoring value = 1.6256e+003

Fig. 8. Tree of solutions for ClustalW (**`TreePW`**) and DescPA (**`TreeHD)`** for 9 proteins

5.1.3 DescPA vs. ClustalW Results

Alignments solutions given by the two options (pairwise for ClustalW and Hellinger distance) of the progressive algorithm are implemented in Matlab™ as follows:

(i) Pairwise distance

SolPW = **multialign** (**seqs**, **TreePW** , **'ScoringMatrix'**, { **'pam150 ',' pam200 ',' pam250'**}); where **TreePW** is a guide tree built using pairwise distance.

(ii) Hellinger distance

SolHD = **multialign** (**seqs**, **TreeHD**, **' ScoringMatrix'**, {**'pam150 ',' pam200 ',' pam250'**}); where **TreeHD** is a guide tree built using the proposed Hellinger distance matrix.

(iii) Results comparison

Figures 9 to 11 show that using the proposed guide tree based on a Hellinger distance gives performance as good as ClustalW in 93% of the cases. To further improve these results, we introduce one iterated local search technique, referred to as 2-opt implemented in Section 5.2.

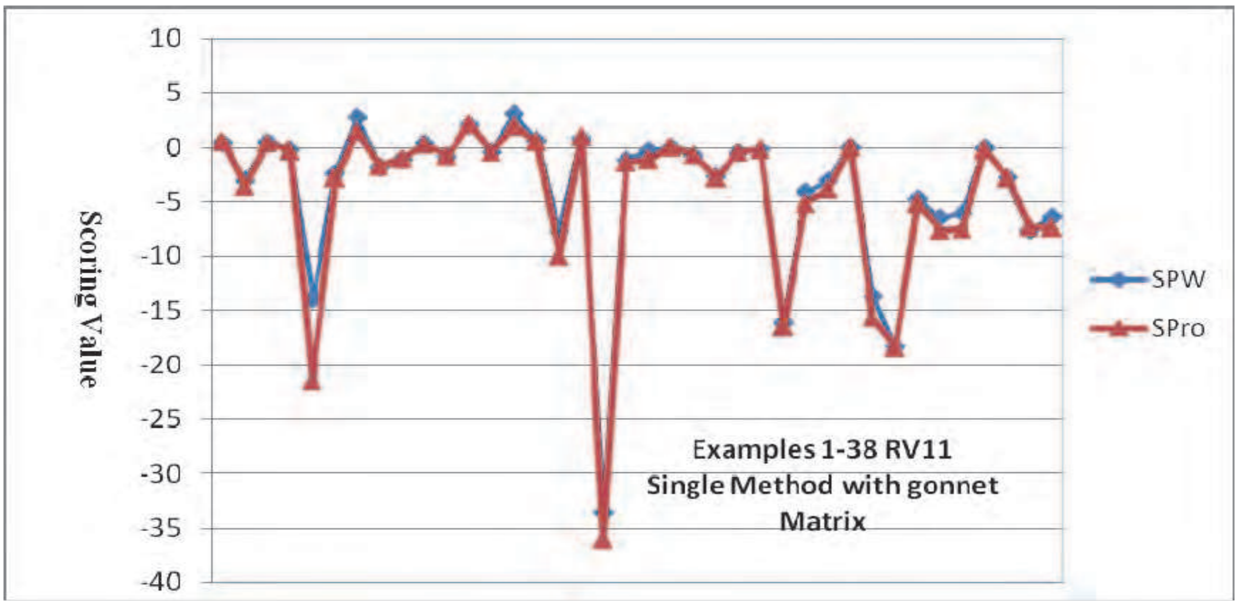


Fig. 9. ClustalW (**SPW**) and DescPA (**Spro**) performance from examples 1-38 (**RV11**)

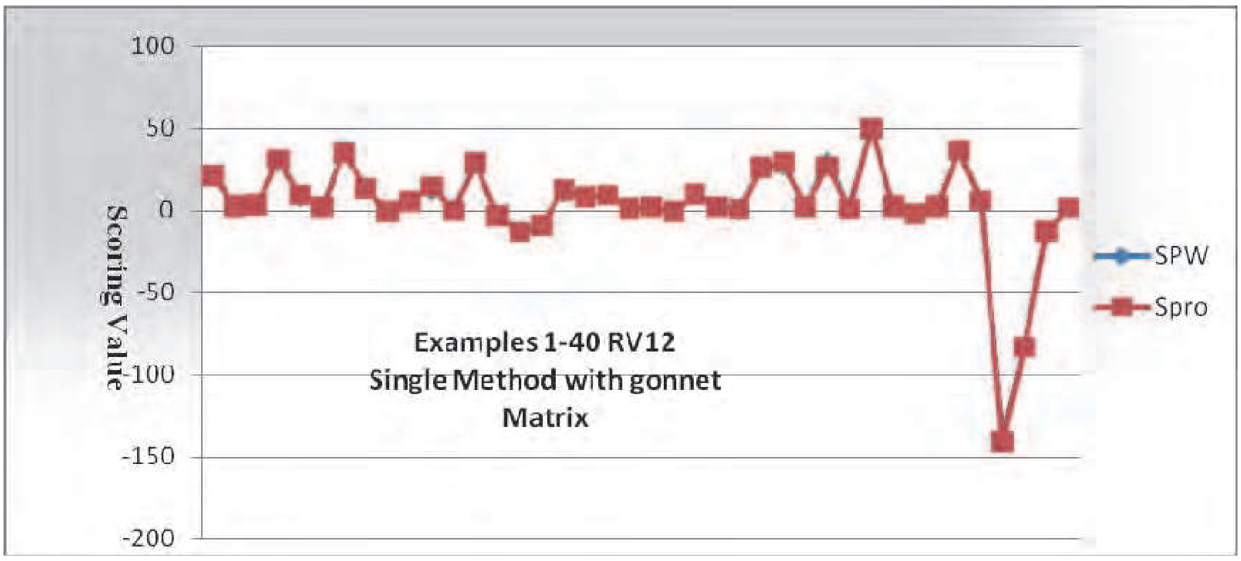


Fig. 10. ClustalW (**SPW**) and DescPA (**Spro**) performance from examples 1-38 (**RV12**)

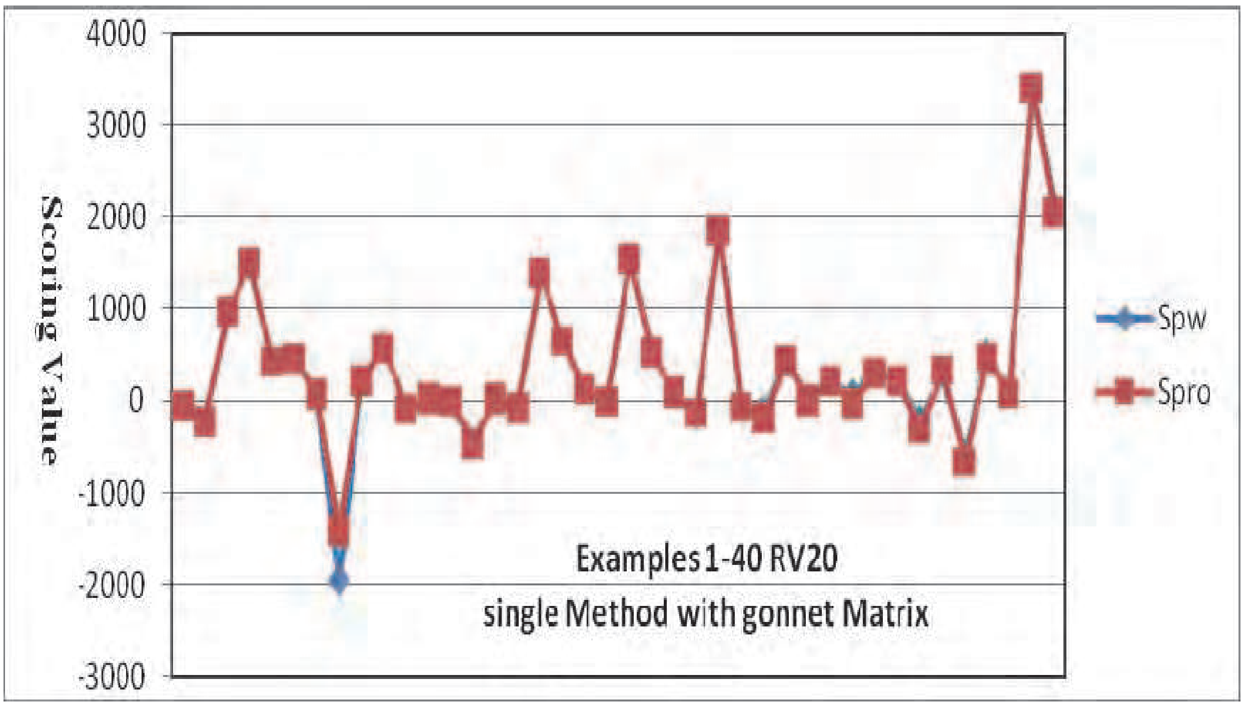


Fig. 11. ClustalW (**SPW**) and DescPA (**Spro**) performance from examples 1-40 (**RV20**)

5.2 DescPA2: Improved results through 2-opt

Figures 12&13 show the improvement on the performance, over different examples of the datasets RV11. There is a clear improvement introduced by the 2-opt algorithm. In Figures 12&13, **SPW** defines the scoring value got using ClustalW, **Spro** gives the scoring value for DescPA and **2-opti** for DescPA2. Despite its simplicity of implementation, the 2-opt algorithm improves the solutions. The final alignments results of DescPA2 are better than those of DescPA and ClustalW.

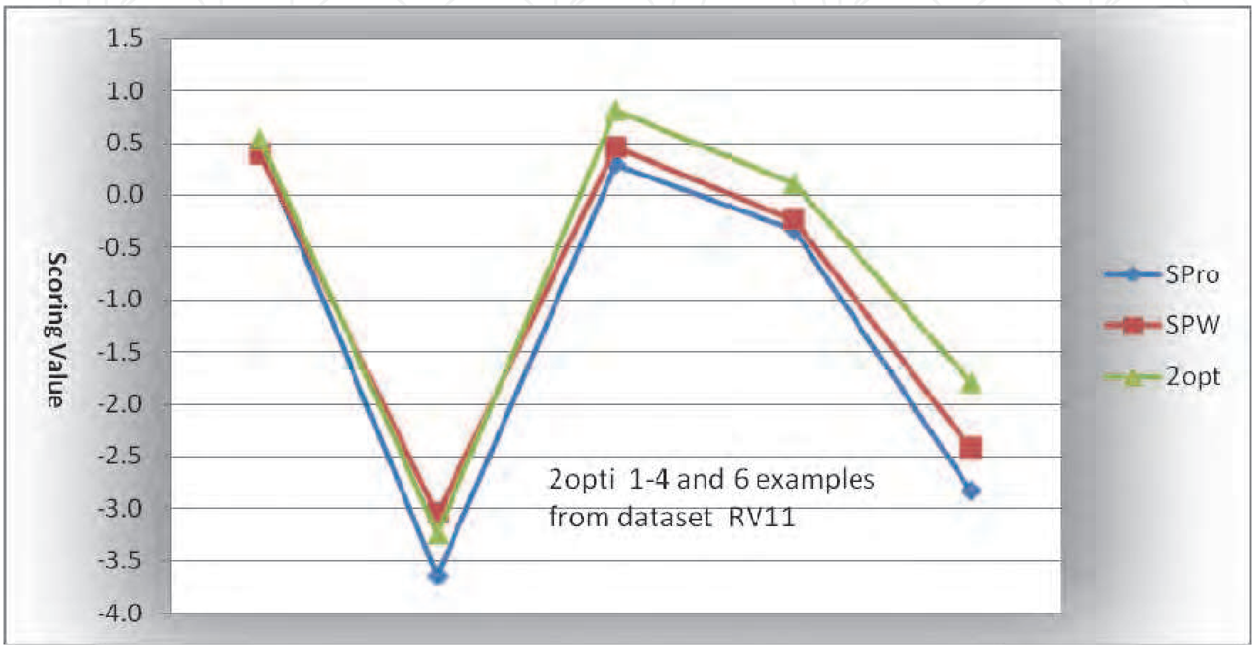


Fig. 12. ClustalW, DescPA and DescPA2 results with 6 examples max from dataset RV11

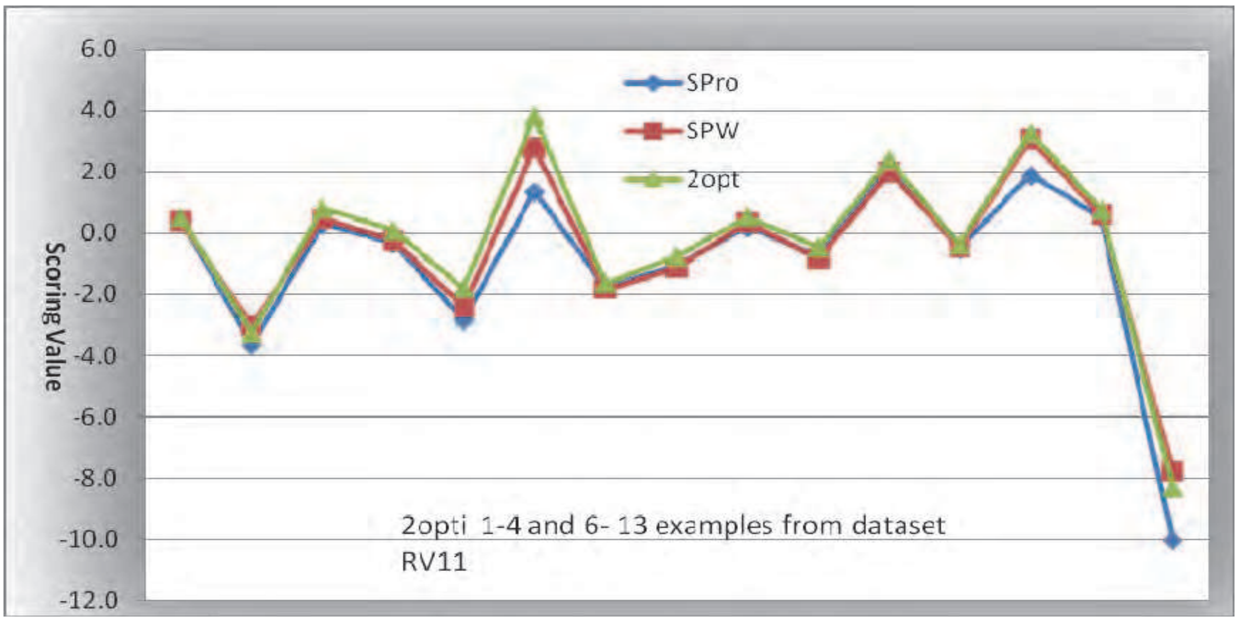


Fig. 13. ClustalW, DescPA and DescPA2 results with 10 examples max from dataset RV11

6. Conclusion

We proposed a modified and hybrid progressive alignment strategy for protein sequence alignment composed of two variants. The first, implemented in DescPA, consists of the modification of the progressive alignment strategy by building a new guide tree based on a Hellinger distance definition. This distance is calculated over a sequences' descriptors; a descriptor being defined for each sequence by its probability density function (PDF). The main feature of this descriptor is its fixed short length (20 for proteins and 4 for DNA) for any sequence length, which mainly impacts positively the computation time for the MSA. The DescPA results of our testing on all the dataset show that the modified progressive alignment strategy is as good as that of ClustalW in 93% of the cases. The second variant, incorporated in DescPA2, is an improvement of the obtained solution using the iterated 2-opt local search. The improvement of the obtained solutions using DescPA2 implementation gives better solutions than DescPA and ClustalW alike - and in all studied cases. As shown, despite its simplicity of implementation, the 2-opt algorithm improves the solutions. However, further improvements are needed. We need, for instance to enhance the actual method to better search through the tree space. For example, we plan to compare DescPA2 with other MSA tools such as hidden Markov models.

7. References

[1] Altschul, S.F. 1991. Amino acid substitution matrices from an information theoretic perspective. *J. Mol. Biol.* (1991), Vol. 219, pp. 555-565, Online ISSN 1460-2059, Print ISSN 1367-4803.

[2] Azim, G.A.; Ben Othman, M. & Abo-Eleneen, Z.A. (2011). Modified progressive strategy for multiple proteins sequence alignment, *International Journal of Computers*. Vol. 5, No.2, (2011), pp. 270-280, ISSN 1998-4308.

[3] Azim, G.A. & Ben Othman, M. (2010). Hybrid iterated local search algorithm for solving multiple sequences alignment problem, *Far East J. of Exp. and Theor. AI,* Vol.5, (2010), pp. 1-17, ISSN 0974-3261.

[4] Bahr, A.; Thompson, J.D.; Thierry, J.C. & Poch O. (2001). BAliBASE (Benchmark Alignment dataBASE): enhancements for repeats, trans-membrane sequences and circular permutations. *Nucleic Acids Res.*, Vol. 29, No. 1, (January 2001), pp. 323-326. Online ISSN 1362-4962, Print ISSN 0305-1048.

[5] Cai, J.J., Smith, D.K., Xia X. & Yuen, K.Y. (2006). *MBEToolbox 2*: an enhanced MATLAB™ toolbox for molecular biology and evolution, *Evol. Bioinf.* Vol. 2, (2006), pp. 189-192, ISSN 1176-9343.

[6] Carrillo, H., & Lipman, D. (1988). The multiple sequence alignment problem in biology. *SIAM J. Appl. Math.* Vol. 48, (1988), pp. 1073–1082, ISSN 0036-1399.

[7] Donoho, D. & Liu, R. (1988). The 'automatic' robustness of minimum distance functionals, *Annals of Stat.*, Vol. 16, (1988), pp. 552-586, ISSN 0090-5364.

[8] Durbin, R. et al. (1998). *Biological Sequence Analysis - Probabilistic Models of Proteins and Nucleic Acids.* Cambridge University Press, ISBN 0521629713, Cambridge, UK.

[9] Giet, L. & Lubrano, M. (2008). A minimum Hellinger distance estimator for stochastic differential equations: an application to statistical inference for continuous time interest rate models, *Comput. Stat. & Data Anal.*, Vol. 52, No. 6, (Feb. 2008), pp. 2945-2965, ISSN: 0167-9473.

[10] Gonnet, G.H.; Cohen, M.A. & Benner, S.A. (1992). Exhaustive matching of the entire protein sequence database. *Science* Vol. 256, (1992), pp. 1443-1445, Online ISSN 1095-9203, Print ISSN 0036-8075.

[11] Higgins D.G., Sharp P.M. (1988). CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. *Gene*, Vol. 73, No. 1, (1988), pp. 237–244, ISSN: 0378-1119.

[12] Henikoff, S. and Henikoff, J.G. 1992. Amino acid substitution matrices from protein blocks. *Proc. Natl. Acad. Sci. U.S.A.Vol.* 89 pp. 10915-10919, Online ISSN 1091-6490, Print ISSN 0027-8424.

[13] Hogeweg, P. & Hesper B. (1984).The alignment of sets of sequences and the construction of phylogenetic trees: an integrated method., *J. Mol. Evol*. Vol. 20, (1984), pp. 175-186, Online ISSN 1432-1432, Print ISSN 0022-2844.

[14] Johnson, D.S. & McGeoch, L.A. (1997). The traveling salesman problem: a case study in local optimization, In Aarts, E.H.L. & Lenstra, J.K. (Eds.) *Local Search in Combinatorial Optimization*, John Wiley and Sons, (1997), ISBN 0471948225, New York, pp. 215-310.

[15] Kumar, S. & Filipski, A. (2007). Multiple sequence alignment: pursuit of homologous DNA positions, *Genome Res.* Vol. 17, (2007), pp. 127-135, ISSN 1088-9051.

[16] 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,* Vol. 262, (1993), pp. 208-214, ISSN 0036-8075.

[17] Loots G. G. & Ovcharenko, I. (2007). *Mulan*: multiple-sequence alignment to predict functional elements in genomic sequences, *Methods Mol. Biol.* 395, (2007), pp. 237-254, ISSN 1064-3745.

[18] Morgenstern, B.; Dress, A.; Wener, T. (1996). Multiple DNA and protein sequence based on segment-to-segment comparison. *Proc. Natl. Acad. Sci. USA* Vol. 93, (1996) pp. 12098-12103, Online ISSN 1091-6490, Print ISSN 0027-8424.

[19] Notredame, C. (2002). Recent progress in multiple sequence alignment: a survey. *Pharmacogenomics*, Vo. 3, No. 1, (2002), pp. 131–144, ISSN 1462-2416.

[20] Roshan, U. & Livesay, D. R. (2006). *Probalign*: multiple sequence alignment using partition function posterior probabilities, *Bioinformatics* Vol. 22, No. 22, (2006), pp. 2715-2721, Online ISSN 1460-2059, Print ISSN 1367-4803.

[21] Saitou N. & M. Nei, (1987). The neighbor-joining method: a new method for reconstructing phylogenetic trees, *Mol. Biol. Evol.*, Vol. 4, No. 4, (1987), pp. 406–425, Online ISSN 1537-1719, Print ISSN 0737-4038.

[22] Thompson, J.D.; Higgins, D.G.; Gibson, T.J. (1994). CLUSTALW: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice, *Nucl. Acids Res.* Vol. 22, (1994), pp. 4673–4680, Online ISSN 1362-4962, Print ISSN 0305-1048.

[23] Thompson, J.D.; Koehl, P.; Ripp, R. & Poch, O. (2005). BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. *Proteins* Vol. 61, (2005) pp.127–136, ISSN 0887-3585.

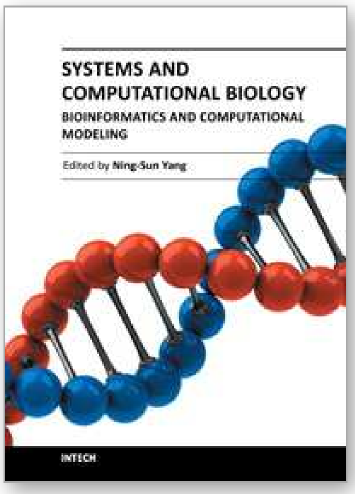
[24] Wang, Y., & Li, K.-B., (2004). An adaptive and iterative algorithm for refining multiple sequence alignment, *Comput. Biol. Chem.* Vol. 28, (2004), pp. 141–148, ISSN 1476-9271.

[25] Wang, L., & Jiang, T. (1994). On the complexity of multiple sequence alignment. *J. Comput. Biol.* Vol. 1, (1994), pp. 337–348, Online ISSN 1557-8666, Print ISSN 1066-5277.

[26] Wang, C. & Lefkowitz, E.J. (2005). Genomic multiple sequence alignments: refinement using a genetic algorithm, *BMC Bioinformatics,* Vol. 6:200, ISSN 1471-2105.

[27] Wheeler, D. (2003). Selecting the right protein-scoring matrix, *Current Protocols in Bioinformatics* (2003) pp. 351-356, Online ISSN 1934-340X, Print ISSN 1934-3396.

[28] Yue, F., Shi J. & Tang, J. (2009). Simultaneous phylogeny reconstruction and multiple sequence alignment, *BMC Bioinformatics* (2009), Vol. 10 (Suppl. 1):S11, ISSN 1471-2105.



Systems and Computational Biology - Bioinformatics and Computational Modeling
Edited by Prof. Ning-Sun Yang

ISBN 978-953-307-875-5
Hard cover, 334 pages
Publisher InTech
Published online 12, September, 2011
Published in print edition September, 2011

Whereas some â€œmicroarrayâ€ or â€œbioinformaticsâ€ scientists among us may have been criticized as doing â€œcataloging researchâ€, the majority of us believe that we are sincerely exploring new scientific and technological systems to benefit human health, human food and animal feed production, and environmental protections. Indeed, we are humbled by the complexity, extent and beauty of cross-talks in various biological systems; on the other hand, we are becoming more educated and are able to start addressing honestly and skillfully the various important issues concerning translational medicine, global agriculture, and the environment. The two volumes of this book present a series of high-quality research or review articles in a timely fashion to this emerging research field of our scientific community.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821