Multiple Sequence Alignment Using Three-Dimensional Fragments

Layal Al Ait, Eduardo Corel, Kifah Tout and Burkhard Morgenstern

***Abstract*—**Background: Dialign is a DNA/Protein alignment tool for performing pairwise and multiple pairwise alignments through the comparison of gap-free segments (fragments) between sequence pairs. An alignment of two sequences is a chain of fragments, i.e local gap-free pairwise alignments, with the highest total score. METHOD: A new approach is defined in this article which relies on the concept of using three-dimensional fragments – i.e. local three-way alignments -- in the alignment process instead of two-dimensional ones. These three-dimensional fragments are gap-free alignments constituting of equal-length segments belonging to three distinct sequences. RESULTS: The obtained results showed good improvments over the performance of DIALIGN.

***Keywords*—**DIALIGN, Multiple sequence alignment, Three-dimensional fragments.

I. INTRODUCTION

OVER the past few decades, major advances in the field of molecular biology have led to huge amounts of DNA and protein sequence data stored in public and private databases. Multiple sequence alignment is a pivotal tool to analyse sequence data. Thus, one of the major fields in bioinformatics is the development of sequence-alignment methods which became the central field of study for many research groups. For a set of two or more DNA, RNA, or amino acid sequences, the goal of sequence alignment is to identify the regions of the sequences that are similar to one another according to some measure and output the sequences with the similar positions aligned in columns. Many softwares were created for this sake. One of which is DIALIGN [5]-[7]-[11]-[12]. Multiple alignments in DIALIGN are based on the concept of using two-dimensional fragments. In our definition, a two-dimensional fragment is a local pairwise gap-free alignment of two of the input sequences. That is, a fragment F consists of two aligned segments (s1,s2) of two of the input sequences. In DIALIGN, every possible fragment is assigned a weight score [6] which is based on the probability of its random occurrence. The overall score of an alignment is then defined as the sum of the weights of the fragments it consists of. Multiple alignments are calculated by performing pairwise alignments on every pair of sequences producing a list of all the fragments which form an optimal pairwise alignment.

These fragments are sorted according to their weight scores and degree of overlap with other diagonals. In this paper we present a new approach which is based on the use of three-dimensional fragments, i.e. local gap-free alignments of three of the input sequences. Such an attempt will catch subtle similarities that become visible when observed simultaneously among many sequences and show positions that are conserved among a whole set of sequences.

II. THE ALGORITHM

1. Finding Three Dimensional Fragments

For a set of N sequences $\{S_1,..., S_N\}$, we align each pair of sequences using DIALIGN's multiple pairwise sequence alignment method. Next, we extend the fragments contained in these pairwise alignments to three-dimensional fragments involving a third sequence S_k as shown in Fig. 1. E.g. for the two-dimensional fragment F between Si and Sj, our job now is to search for the best matching fragment in S_k. To be precise, if F consists of the segments s1 and s2, we are looking for a segment s3 from sequence S_k such that the sum of the weights of the fragments (s1,s3) and (s2,s3) is maximized [2,3,10]. We use a sliding window with size S= length(F).

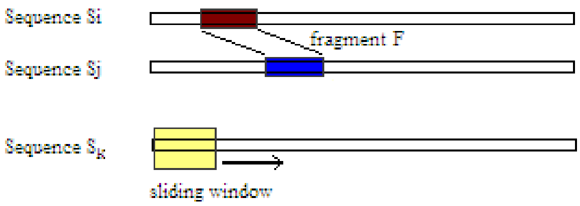

Fig. 1 Searching for the best matching segment in S_k for a given fragment F from sequences Si and Sj using a sliding window which scans the whole sequence.

This process is repeated for all the fragments between Si and Sj, and for every sequence S_k in the remaining N-2 sequences.

In Fig.2, the fragments from sequence pair (Si, Sj) are considered. Each of these fragments is extended to a segment from S_k since for every aligned fragment between Si and Sj, a

[1]Layal Al Ait e-mail: layal@ gobics.de
[2]Eduardo Corel e-mail: eduardo@ gobics.de
[3]Kifah Tout e-mail: ktout@ul.edu.lb
[1]Burkhard Morgenstern e-mail: burkhard@ gobics.de

matching fragment in S_k is allocated. This is how three-dimensional fragments are found.

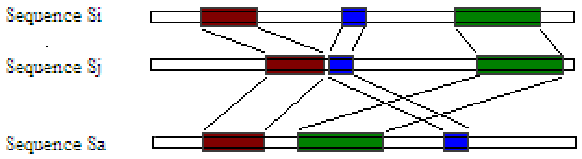

Fig. 2 Three “three-dimensional fragments” formed between Si, Sj and $S_{k.}$

At this point, two threshold values T and T’ are introduced. According to the example, a three dimensional fragments group is taken into consideration if and only if the weight of fragment Si - Sj is higher than T, and the weights of the two fragments of Si - S_k and Sj - S_k are higher than T’. This is important since fragments with low scores when aligned could destroy the alignment; they are considered to be noisy data.

2. Calculating The Multiple Alignments Using Three-Dimensional Fragments

The constructed fragments indicate what triples of positions should (if possible) be aligned. However, not all of the three-dimensional fragments calculated as described above can be included in a final alignment of DIALIGN. A set of fragments can be inconsistent, i.e. it may not be possible to include all of them simultaneously into one multiple alignment. To find a consistent set of fragments, we use our three-dimensional fragments as so-called anchor points for DIALIGN, see [8]. DIALIGN then selects a consistent subset of these anchor points using its greedy algorithm for multiple alignments.

The potential inconsistencies are detected in the standard DIALIGN algorithm by greedily [1] inserting the fragments, and rejecting any fragment whose inclusion turns out to be impossible due to previously accepted fragments (*inconsistency*) [4]. It is a quite well known problem that the greedy approach thus used can be put at fault when one single inconsistent fragment (arising for instance from a random or biologically unsignificant similarity presenting however a good similarity score) prevents the inclusion of a whole family of weaker but overall conserved similarities shared by a large fraction of the sequences. Therefore, we recently introduced an alternative approach to resolve consistency conflicts in multiple sequence alignment [14]. To avoid the above phenomenon, the three-dimensional fragments are processed by a further algorithm before being fed to DIALIGN.

Two kinds of specific inconsistencies are successively investigated, and translated into a graph-theoretical setting. Trying to align the triplets of positions included in the fragments (that we call "elementary triplets") can lead step by step to attempting to align two positions belonging to the same sequence. This configuration is avoided by discarding a subset of similarities among the ones implied by the fragments. Namely, we construct a graph encoding the elementary triplet relations, and cluster it into densely related groups of positions containing at most one element in each sequence. In this way, the step-by-step alignment hints are restricted to groups of positions that can independently fit into a multiple alignment, and that we call *partial alignment columns*. In a second step, the *order* between these partial alignment columns is taken into account, and encoded as a directed graph. The inconsistencies appear as cycles in this graph, and the algorithm proceeds to remove positions from the partial alignment columns that are responsible for the cycles appearance.

After this algorithm has been applied, the remaining aligned positions fit without further restrictions in a multiple alignment. They can then be included into DIALIGN as *anchor points.*

III. Time Complexity

The explained algorithm has a rather high time complexity. Considering every couple of sequences, and then aligning all the remaining N-2 sequences takes O(N^3) time and will therefore consume a lot of time. For example, running the algorithm on a file containing 180 sequences may take a complete day or more to finish the whole execution on an ordinary computer. That is why an option is added. The user now can choose the number of sequences K to be aligned with every two-dimensional fragment. This way, the software searches for the K best sequences to be aligned with each two-dimensional fragment. This is done by finding the K sequences with the highest similarity w.r.t. the respective pair of sequences to be aligned with. To be more specific, this is achieved through the following set of steps:

- For every input sequence A, and for every sequence B different from A, loop through the list of input fragments from the pairwise alignments and save the value W (which is the sum of all the weights of the fragments between A and B) in a list. Choose for A the K best sequences, i.e. the K sequences having the highest value of W. During the alignment process, Sequences S and S’ are considered only if S’ is one of the K best sequences of S. Then, for the couple S and S’, align S’’ with them if and only if S’’ is one of the K best sequences of S or K best sequences of S’.

The last step is to enter the set of anchor groups along with the original sequence file to DIALIGN [8,9] to make the final alignment.

IV. TESTING

1. *BAliBASE*

Testing the quality of the aligner and its ability to produce biologically correct alignments is done using BAliBASE [12]. This database is a benchmark alignment database for the evaluation of multiple protein alignment programs. BAliBASE consists of manually refined multiple sequence alignments containing core blocks of relatively high sequence conservation. Sequence sets in BaliBASE are categorized according to sequence length, similarity and other criterions. It contains six datasets named: RV11, RV12, RV20, RV30, RV40 and RV50.

2. *Parameters*

To test our program, we varied the following parameters:

- Parameter 1: the number K of sequences to which a two-dimensional fragment is compared.
- Parameter 2: the first threshold T
- Parameter 3: the second threshold T'

Testing is carried out in this study using 15 different values for the thresholds and 2 different values for K, namely 4 and 10. The threshold values applied in the testing vary between 0 and 6 for T and between 0 and 20 for T'.

Due to space limitations, only the testing results using the threshold values 1.0-1.0 and 3.0-20.0 and the number of best matches = 4 are displayed.

3. *Results and Discussion*

The following tables show the results of applying the aligner on all the sequences of the BAliBASE benchmark. Table.1 summarizes the results obtained by using the threshold values 1.0-1.0. The first column corresponds to the testing set, The second column represents the percentage of the sequences having a score higher than that produced by DIALIGN, while the third one represents the percentage of the sequences having a score equal to those produced by DIALIGN. The forth column represents the total of the second and third columns.
Table.2 summarizes the results obtained by using the threshold values 3.0-20.0.

TABLE I
TESTING RESULTS USING THRESHOLD VALUES 1.0-1.0.

Test set	Higher (%)	Equal (%)	Total (%)
RV11	28.94736	23.68421	52.63158
RV12	13.63636	0.0	13.63636
RV20	14.63414	2.43902	17.07317
RV30	13.33333	3.33333	16.66668
RV40	10.20408	2.04081	12.24489
RV50	6.25	0.0	6.25

TABLE II
TESTING RESULTS USING THRESHOLD VALUES 3.0-20.0.

Test set	Higher (%)	Equal (%)	Total (%)
RV11	0.0	2.631579	2.63157
RV12	11.36363	36.36363	47.72727
RV20	24.39024	29.26829	53.65853
RV30	36.66666	36.66668	73.33333
RV40	22.44897	38.77551	61.22448
RV50	37.5	31.25	68.75

Running the aligner on RV11 produced almost the best results for the threshold values 1.0-1.0, since 28.94739% of the alignments had scores higher than DIALIGN with an average value of 0.45165. But this value is still lower than the average score of DIALIGN which is 0.46186. When using thresholds values of (3.0,20.0), non of the produced alignments gave results better than DIALIGN, but we can observe that the average score is somehow equal to that of DIALIGN, this is due to the fact that the threshold values are high, and non of the three-Dimensional fragments had weight values higher than those thresholds.
The results of RV30 for the threshold values 3.0 and 20.0 shows that 36.66668% of the results were better than DIALIGN, and the average score was slightly higher than that of DIALIGN too. For RV20, RV40 and RV50, applying the thresholds 3.0 and 20.0 gave a total greater than 50%, which shows that the aligner is functioning better than DIALIGN on the set as a whole.
As it is observed in table.3, the total average score produced is slightly less than or equal to the score produced by DIALIGN, and in the case of RV30, using thresholds 3.0-20.0, it was slightly higher. Fig. 3 provides a clearer overview for the performance of the three-dimensional aligner.

TABLE III
COMPARING THE AVERAGE SCORE OF THE ALIGNER TO THAT OF DIALIGN

Test set	Thresholds	Score	DIALIGN score
RV11	1.0 – 1.0	0.45165	0.46186
	3.0 – 20.0	0.46186	
RV12	1.0 – 1.0	0.78040	0.84897
	3.0 – 20.0	0.83572	
RV20	1.0 – 1.0	0.80646	0.84790
	3.0 – 20.0	0.84180	
RV30	1.0 – 1.0	0.59996	0.64256
	3.0 – 20.0	0.64266	
RV40	1.0 – 1.0	0.73312	0.78751
	3.0 – 20.0	0.78279	
RV50	1.0 – 1.0	0.70343	0.77050
	3.0 – 20.0	0.7645	

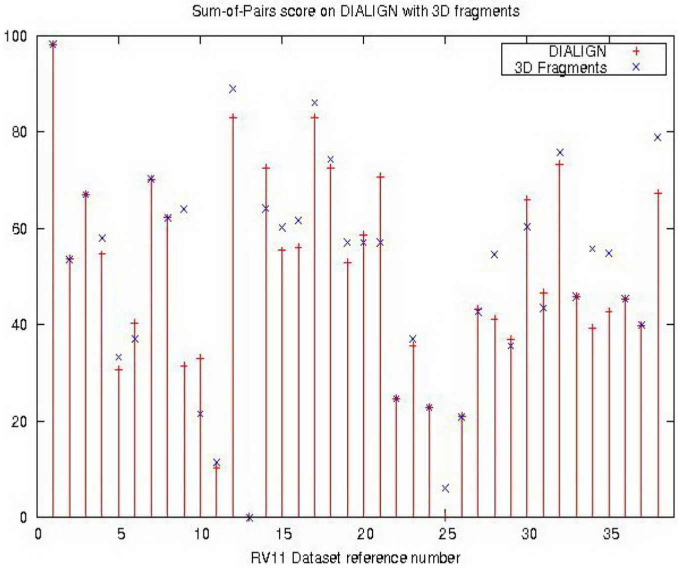

Fig. 3 Comparision of the scores produced by DIALIGN and by the 3D aligner on RV11 (using thresholds 1.0-1.0).

In Fig.3, a comparison is made between DIALIGN and the three-dimensional aligner; the x-axis corresponds to the different sequence sets of RV11 which are 38 sequence sets, and the y-axis represents the score. This graph shows that the performance of the three-dimensional aligner is better than DIALIGN in most of the cases, and in some cases they produced the same score.

4. Conclusion and Future Work

In this project we have presented a new approach for DIALIGN which relies on using three-dimensional fragments in the alignment process instead of using the two-dimensional ones used so far by DIALIGN. Jumping from two to three dimensional fragments helped in catching subtle similarities that were not visible when comparing two sequences with each other.

It is quite obvious that although, on average, the three dimensional aligner did not perform better than DIALIGN, it did produce some good results which were better than those produced by DIALIGN in many test cases. This can imply that there exist some important fragments which should be aligned, but DIALIGN is not including them in the alignment procedure (maybe due to their low weight score or because of the inconsistency produced if they are aligned) and if they are included, DIALIGN would perform much better and produce more correct results. Further studies will be carried out to test if a certain fragment, if included in the optimal alignment, would affect the whole alignment positively and not destroy it.

ACKNOWLEDGMENT

We thank Dr. Thomas Lingner for exchanging his interesting ideas and thoughts.

REFERENCES

[1] Abdeddaim,S. and Morgenstern,B. (2001) Speeding up the DIALIGN multiple alignment program by using the ‘greedy alignment of biological sequences library’ (GABIOS-LIB). Lect. Notes Comput. Sci., 2066, 1–11.

[2] Dayhoff,M.O., Schwartz,R.M. and Orcutt,B.C. (1978) A model of evolutionary change in proteins. Atlas Protein Seq. Struct., 6,345–362.

[3] Henikoff,S. and Henikoff,J.G. (1994) Protein family classification based on searching a database of blocks. Genomics, 19, 97–107.

[4] Morgenstern,B., Dress,A., Werner,T., (1996) Multiple DNA and protein sequence alignment based on segment-to-segment comparison. Proc. NAtl Acad. Sci USA, 93, 12098-12103.

[5] Morgenstern,B., French,K., Dress,A., Werner,T., (1998) DIALIGN: Finding local similarities by multiple sequence alignment. Bioinformatics, 14, 290-294.

[6] Morgenstern,B., Atchley,W.R., Hahn,K. and Dress,A. (1998) Seqment-based scores of pairwise and multiple sequence alignments. In Glasgow,J., Littlejohn,T., Major,F., Lathrop,R, Sankoff,D. and Sensen,C. (eds), Proceedings of the Sixth International Conference on Intelligent Systems for Molecular Biology. AAAI Press, Menlo PArk, CA,pp. 115-121.

[7] Morgenstern,B. (1999) DIALIGN 2: improvement of the segment-to-segment approach to multiple sequence alignment. Bioinformatics, 15, 211-218.

[8] Morgenstern,B., Werner,N., Prohaska,S.J., Steinkamp,R., Schneider,I., Subramanian,A.R., Stadler,P.F. and Weyer-Menkhoff,J. (2005) Multiple Sequence alignment with user-defined constraints at GOBICS. Bioinformatics, 21, 1271–1273.

[9] Morgenstern,B., Prohaska,S.J., Po¨ hler,D. and Stadler,P.F. (2006) Multiple sequence alignment with user-defined anchor points. Algorithms for Molecular Biology, 1, 6.Corel,E., Pitschi,F. and Morgenstern,B. (2010) A min-cut algorithm for the consistency problem in multiple sequence alignment. Bioinformatics., 26, 1015–1021.

[10] Needleman,S. and Wunsch,C. (1970) A general method applicable to the search for similarities in the amino acid sequences of two proteins. J. Mol. Biol., 48, 443-453.

[11] Subramanian1,A.R., Weyer-Menkhoff,J., Kaufmann,M., Morgenstern,B., (2005) DIALIGN-T: An improved algorithm for segment-based multiple sequence alignment. BMC Bioinformatics.

[12] Subramanian1,A.R., Hiran,S., Steinkamp,R., Meinicke,P.,Corel,E., Morgenstern,B., (2010) DIALIGN-TX and multiple protein alignment using secondary structure information at GOBICS. Nucleic Acids Research, 38, 19-22.

[13] Thompson,J.D., Koehl,P., Ripp,R. and Poch,O. (2005) BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark. Proteins, 61, 127–136.

[14] E.Corel, F. Pitschi, B. Morgenstern (2010) A min-cut Algorithm for the Consistency Problem in Multiple Sequence Alignment Bioinformatics 26, 1015-1021, doi:10.1093/bioinformatics/btq082.