E-PROBCONS: Enhanced PROBCONS for Multiple Sequence Alignment

Eman M. Mohamed, Hamdy M. Mousa, Arabi E. keshk
Computer Science Dept., Faculty of Computers and Information, Menoufia University, Egypt
{ eman.mohamed@ci.menofia.edu.eg, hamdimmm@hotmail.com, arabikesk@yahoo.com.}

Abstract— the perfect alignment between three or more sequences of protein, RNA, or DNA, is a very difficult task in Bioinformatics. There are many techniques for the alignment of multiple sequences. Many techniques enlarge speed and do not have a concern with the accuracy of the resulting alignment. However, other techniques heighten accuracy and do not have a concern with the speed. The vital goals of any technique are (a) reducing memory and execution time requirements, and (b) increasing the accuracy of multiple sequence alignment on large-scale datasets. PROBCONS is a multiple protein sequence alignment (MPSA) tool that achieves the most expected accuracy, but it has a time-consuming problem. To solve this problem and enlarging the accuracy of the MPSA, E-PROBCONS is proposed to enhance the PROBCONS tool. E-PROBCONS cluster the large multiple protein sequences into structurally similar protein sequences. Then PROBCONS MPSA tool will be performed in parallel on the Amazon Elastic Cloud (EC2). The proposed approaches are more suitable for large-scale data sets and short sequences. Comparing with algorithms (e.g., PROBCONS, KALIGN, and HALIGN I), provided more than 50% improvement in terms of average sum of pair alignment scores (SPscores) and reduce the execution time for producing the alignment result. The proposed approaches are implemented on big data framework Hadoop Map-Reduce platform to improve the scalability with different protein datasets.

Keywords—Bioinformatics, Multiple sequence alignment, Protein features, PROBCONS.

I. INTRODUCTION

Alignment is the process of putting at least two amino acids in the same columns to achieve the maximum level of similarity, this similarity indicates the relationship between sequences [1]. The alignment algorithms, classified into two categories local and global alignment, global uses the entire sequences expand the quantity of matched residues, for example, the Needleman-Wunsch algorithm.

```
F G K S  T K Q T G K G
|        |     | | |
F N A T A K S A G K G
```

But Local algorithms maximize the alignment of similar sub-regions, for example, the Smith-Waterman algorithm.

```
_ _ _ _ _ _ _ _ _ F G K G _ _ _ _ _ _ _ _ _
                  | | |
_ _ _ _ _ _ _ _ _ F G K T _ _ _ _ _ _ _ _ _
```

Multiple sequence alignment (MSA) is contained more than pairwise sequences. MSA is aligned simultaneously obtained by inserting gaps (-) into sequences [2, 3]. An example of MPSA is presented in figure 1. To get the ideal protein MSA, there are many MSA methods. MSA methods are classified into dynamic programming (DP) [4] and heuristic [5] as shown in figure 2. DP gives the optimal MSA. The heuristic techniques divide into progressive [6], iterative [7], and probabilistic [8] technique. Heuristic MSA produces an approximate solution.

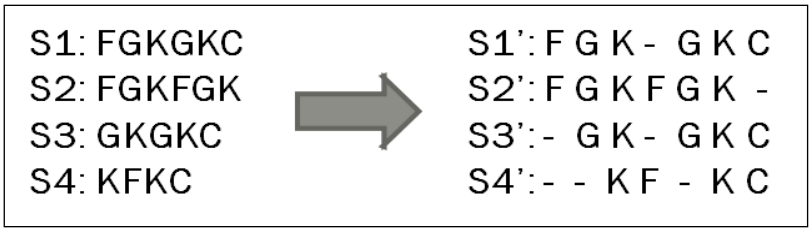

Fig. 1. MSA example, with four protein sequences.

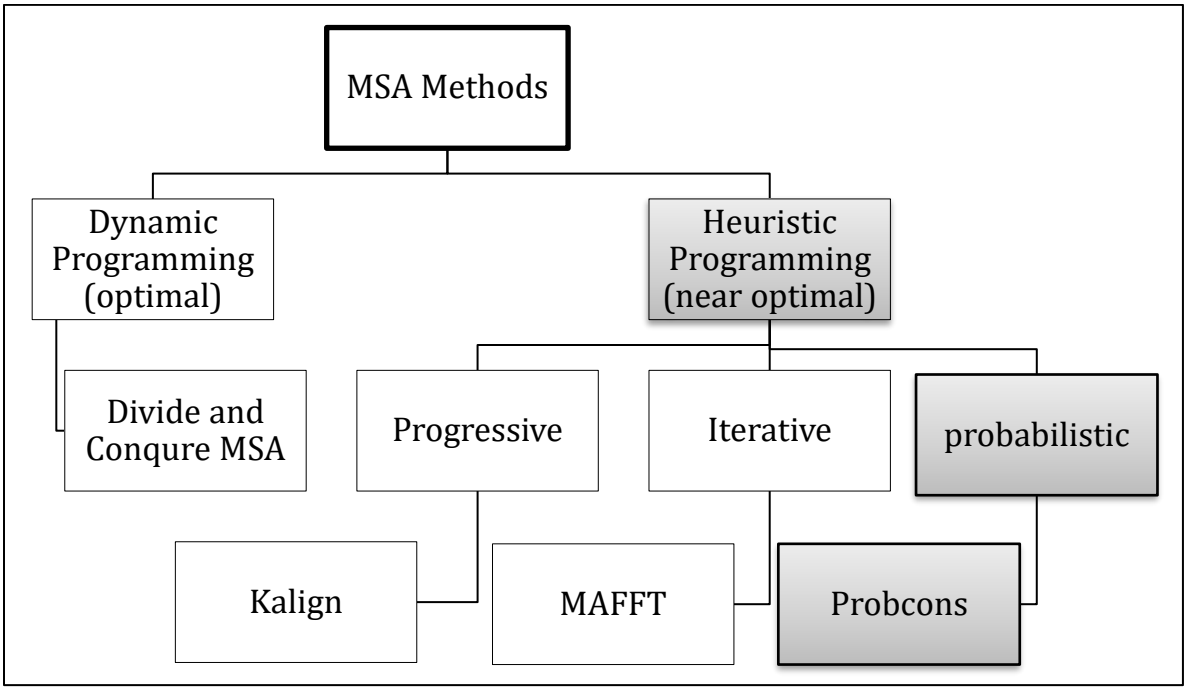

Fig. 2. MSA Methods Classification.

DP gives the optimal MSA, but it is more time consuming, so DP used to align a few sequences. The most popular algorithm for DP is called divide and conquer algorithm (DCMSA) [9, 10]. DCMSA algorithm is introduced by Stoye, which protein sequences are divided into two regions, then into four regions and therefore to eight regions, and so on until the sequences are shorter to be predetermined or considered small enough. For optimal alignment, the subsequences are then aligned and in the last step, the alignment is assemblies. Therefore, aligning multiple long sequences is divided into several smaller alignment tasks. The main problem in the DCMSA algorithm is how to determine the position for cutting of each sequence.

The heuristic techniques divide into a progressive, iterative, and probabilistic technique. Progressive MSA was implemented at three main steps. The first step is calculating the pairwise score and convert them to the distance matrix. The pairwise calculation is done using DP algorithms. The second step is constructing a guide tree from a distance matrix using clustering techniques. Finally, in the last step, align the sequences in their order of the tree. The big advantage of progressive MSA is used to align a large number of biological sequences. However, it produces a near-optimal alignment, in which the final alignment depends on the order of aligned pairwise. The most popular programs for progressive MSA is the Clustal family [11, 12] (ClustalX, ClustalW, and Clustal-Omega) and KAlign family (Kalign1, Kalign2, and Kalign-LCS) [13, 14].

Iterative MSA makes an initial alignment of multiple sequences based on some progressive MSA algorithms and then iteratively improve the alignment result to achieve the ideal MSA. For example MAFFT [15] and MUSCLE [16]. MUSCLE tries to make initial MSA as fast as possible and then generate a log-expectation score to perform profile to profile alignment. Unfortunately, previous methods, highly depend on the initial MSA or initial alignment stages.

Finally, for probabilistic procedures, PROBCONS [8] (probabilistic-CONSistency-based multiple-alignments-of-amino-acid-sequences) is a tool for creating MPSA dependent on probabilistic consistency. PROBCONS has achieved the most raised accuracy's of all MPSA techniques as of recently. The probabilistic consistency technique has been utilized by PROBCONS for different protein sequences. In any case, PROBCONS cannot be legitimately utilized for multiple template stringing when proteins under thought are distantly-related, which PROBCONS does not utilize much protein structure data in creating a probabilistic MSA; PROBCONS discard gaps-penalty since it is extravagant to appraise the likelihood of a gap. Gap-penalty ignoring is good for close protein homologs, but it may affect accuracy when protein sequences are indirectly related. So the fundamental issue in the PROBCONS tool is that cannot utilize structure protein data. PROBCONS is not truly fine at distantly-related protein sequence alignment because PROBCONS disregards gap-penalty, to accomplish sensible computational productivity [8]. Different highlights of the tool incorporate the utilization of using twofold of affine insertion penalties, guide tree computation through semi probabilistic clustering, iterative refinement, and unsupervised Expectation-Maximization (EM) preparing of gap parameters. PROBCONS gives a sensational improvement for MPSA accuracy over existing tools. It accomplishes the most astounding scores on the BALIBASE [17] benchmark of any presently realized MPSA tools.

In this paper, clustering the large scale protein sequences based on ten biology protein features. These features classify similar protein sequences to reduce the execution time of the PROBCONS tool. Some of the features related to protein secondary structure (PSS) prediction [18, 19]. To complete the set of biological features, the clustering of an amino acid (AA) is represented [20, 21]. Finally, to achieve accurate alignment, we classify large protein sequences based on the longest common subsequence (LCS) [22]. After that, Apply PROBCONS multiple sequence alignment tools in parallel for each cluster on the Amazon EC2 cloud computing platform [23].

The organization of this paper is as follows. The related work is introduced in section II. Section III explains the proposed MPSA. The implementation environment is viewed in section IV. The simulation and experimental results are discussed in section IV.

II. RELATED WORK

The clustering of protein sequences had an important role in Bioinformatics ' real-world application. The clustering is used to understand protein function and protein structure and to know the structure or function of a new protein sequence. In biological research, previous protein clustering methods are introduced at different categories [24]. The important methods are feature-based clustering and sequence distance based on clustering or using HMM [25] or other statistical methods for clustering.

There are several MSA tools with different attributes, but no single MSA tool can always achieve the highest accuracy with the lowest execution time for all test cases. The parallelization approach is focused to decrease memory and execution time. More different parallelization strategies are implemented to reduce time. Most of the existing MSA parallelization approaches is implemented on multi-core computers [26] or mesh-based multiprocessors [27, 28] or multithreading [29] or MPI (multiprogramming interface) [30] or Hadoop [31] or spark [32] or GPU [33, 34] or clouds [23].

FAMSA [35], one of the progressive calculation intended for quick and precise alignment of thousands of sequences of protein. Its highlights incorporate the use of the longest common subsequence measure for deciding pairwise likenesses, a novel strategy for gap costs assessment, and another iterative refinement conspire. Critically, its usage is exceptionally enhanced and parallelized to benefit as much as possible from present-day PC stages.

MSACompro [36] is another productive and dependable numerous protein MSA. It consolidates anticipated optional structure, relative dissolvable availability, and buildup contact data into the as of now the most exact back likelihood-based MSA strategies. It utilized various stringing execution on a 32 CPU center machine.

MSAProbs [37, 38] is another and reasonable various protein MSA structured by consolidating a pair HMM and a partition function to calculate posterior probabilities. It likewise explores two basic bioinformatics procedures, to be specific weighted probabilistic consistency change and weighted profile-profile arrangement, to accomplish high arrangement precision. Moreover, it is improved for present-day multi-center CPUs by utilizing a multi-strung plan to reduce execution time.

With the rapid growth of biological datasets, MSA techniques must be efficient for large-scale biological data sets. Large-scale MSAs has also the challenge of time and space consuming. Therefore, parallelization is a key approach for decreasing the time execution [39]. There are numerous strategies for alignment with more than two sequences.

Some of them minimize time and do not matter by the accuracy of the resulting alignment. Likewise, many strategies maximize accuracy and do not concern with the running time. Decreasing memory and execution time necessities and increasing the MSA accuracy on large-scale datasets are the crucial intention of any technique [33].

In [40] assesses groups with MSA tools of BALIBASE datasets for accuracy, execution time, impacts of sequence length, and sequence number. The Results demonstrated that the PROBCONS accuracy is the highest for all the examined MSA tools, yet it was a moderate, slow tool and PROBCONS has no more than 1000 sequences in the alignment.

Cloud computing is a model that enables flexible computing as a service utility. It provides a scalable infrastructure to compute with storage and other computing issues. There are many cloud providers, which a different service has been offered to users on the internet. Cloud computing has many advantages, which users do not worry about the computing future needs such as maintenance, resources, availability, and reliability issues. Cloud users only pay for used resources types and time. As a result, the cloud platform is an important solution for big data analysis, especially in the Bioinformatics research field. The Cloud model solves the storage and computational issues for large-scale data analysis. So biology clients don't need to have a high capacity computer for biological data analysis. The cloud provides a high availability data and also provide on-demand powerful computers. Biologists only need the internet with high speed to connect with cloud services [23]. For example, Cloud-Coffee [41] is a parallel implementation of T-Coffee but in a different Approach.

In this paper, PROBCONS is enhanced. It is an MPSA tool that achieves the most expected accuracy, but it has a time-consuming problem. E-PROBCONS is the proposed enhancement of PROBCONS. E-PROBCONS solve the time problem and enlarging the accuracy of the MPSA, in which the large multiple protein sequences are clustered into structurally similar protein sequences. Then PROBCONS MPSA tool is performed in parallel on the Amazon Elastic Cloud (EC2).

III. PROPOSED ALGORITHM

Various feature protein sequences may be applied such as Amino-acids clustering [20, 21], average chemical shift [42], k-mer classification [43], and secondary structure prediction [18, 19]. All these features are strongly affected by sequence, accuracy, and similarity.

The fundamental problem with large-scale sequence alignment is time-consuming. Most current MSA tools are not producing the highest accuracy with less time execution and not suitable for every dataset. MSA with a growing number of sequences (more than 100) is a time consuming and become a big problem to solve. To solve the large-scale problem, the proposed has clustered the protein sequences based on some biological features. After that, apply the PROBCONS MPSA alignment tool in parallel. Finally, merge the alignment results for each cluster.

Therefore, first, divide protein sequences into groups based on some biological features. So at first let $S= S_1, S_2, S_3, \ldots, S_N$, which S contains the N Protein sequences.

TABLE I. PROTEIN clustering FEATURES

	Feature Name
Related to Sequence (*FLCS*)	Number Of Sequences
	Average Length
	Reference Subset
	Data Type (DNA, Protein)
	Longest Common Subsequence
Related to Secondary Structure (*FSS*)	**α-Helix**
	B- Sheet
	Coil
Related to Amino Acids (FAA)	**Polar Uncharged Amino Acids**
	Nonpolar Aliphatic Amino Acids
	Basic Positively Charged Amino Acids
	Aromatic Amino Acids
	Negatively Charged Amino Acids
	BasicKR

- The sequence S_i is placed in the first group. Let *M* be the group number and S_i belongs as a center sequence in *M* group.
- Then, the second sequence *Sj* is compared based on some biological features to S_i.
    - The second sequence would belong to the *M* group if the result is more than a threshold;
    - Otherwise, it would form a new group.
- For each group, apply a PROBCONS MSA tool.
- Merge between groups progressively to retrieve MSA.
- Store MSA as FASTA file.

Protein sequences may have similar functions and structures, so in this case, it has high similarity. So in the feature selection phase classifying large protein sequences based on the LCS, related to PSS prediction (β- strand, α-helix, and coil structures), and related to amino acid (AA) representation (Aromatic AA, Basic KR AA, Nonpolar AA, Negative Polar Charged AA, Positive Polar Charged AA, and Polar UN Charged AA). The list of features represented in Table I.

A. Clustering related to the sequence

This feature clustering is based on LCS length between two different protein sequences. We define the similarity-based LCS for S_i and S_j as follows:

The LCS is defined by the following formula:

$$LCS(i,j) = \begin{cases} 0 & if\ \ i = 0,\ j = 0 \\ LCS(i-1, j-1) + 1 & if\ ai = bj \\ \max\left(LCS\big(i, j-1, LCS(i-1, j)\right) & other\ wise \end{cases} \quad \textbf{(1)}$$

Where *a* and *b* are two sequences, *i* and *j* describe row and columns. The cells in the first row and column are filled with zero for initialization. $i = 1, 2, \ldots, m$ and $j = 1, 2, \ldots, n$. The number of rows and columns in *LCS* are $m+1$ and $n+1$, respectively, Whereas the cell *LCS (i, j)* is the element in the *LCS* table at row *i* and column *j*. The *LCS* table stores numbers, which correspond to the actual length of the *LCS*. After filling the *LCS* table, the lower right cell in the table contains the length of the *LCS*. The longest common subsequence can be found by tracing back from the cell at *LCS (m, n)*. Each time a match is found, it is appended to the longest common subsequence and a movement is made to cell *LCS (i -1; j -1)*. When the symbols do not match, a movement is made to the cell with max *((LCS (i -1, j), LCS (i, j-1))* to find the next match.

$$S_{LCS} = \frac{LCS(i,J)}{Avg(i,j)} * 100 \quad \textbf{(2)}$$

B. Amino-acids clustering

The amino acid is composed of the 20 amino-acid types. We classify the amino acids as:

- Nonpolar Uncharged Amino Acids (*NPUAA*) [G, A, P, V, L, I, M] Percentage,
- Polar Aliphatic Amino Acids (*PUAA*) [S, T, C, N, Q] Percentage,
- Basic Positively Charged Amino Acids (*PCAA*) [K, R, H] Percentage,
- Aromatic Amino Acids (*AAA*) [F, W, Y] Percentage,
- Negatively Charged Amino Acids (*NCAA*) [D, E] Percentage and
- Basic KR residues (*BKR*) [K, R] Percentage as shown in figure 3.

Amino-acids are clustered into six categories to reflect the information of sequence order and accuracy based on Amino acids.

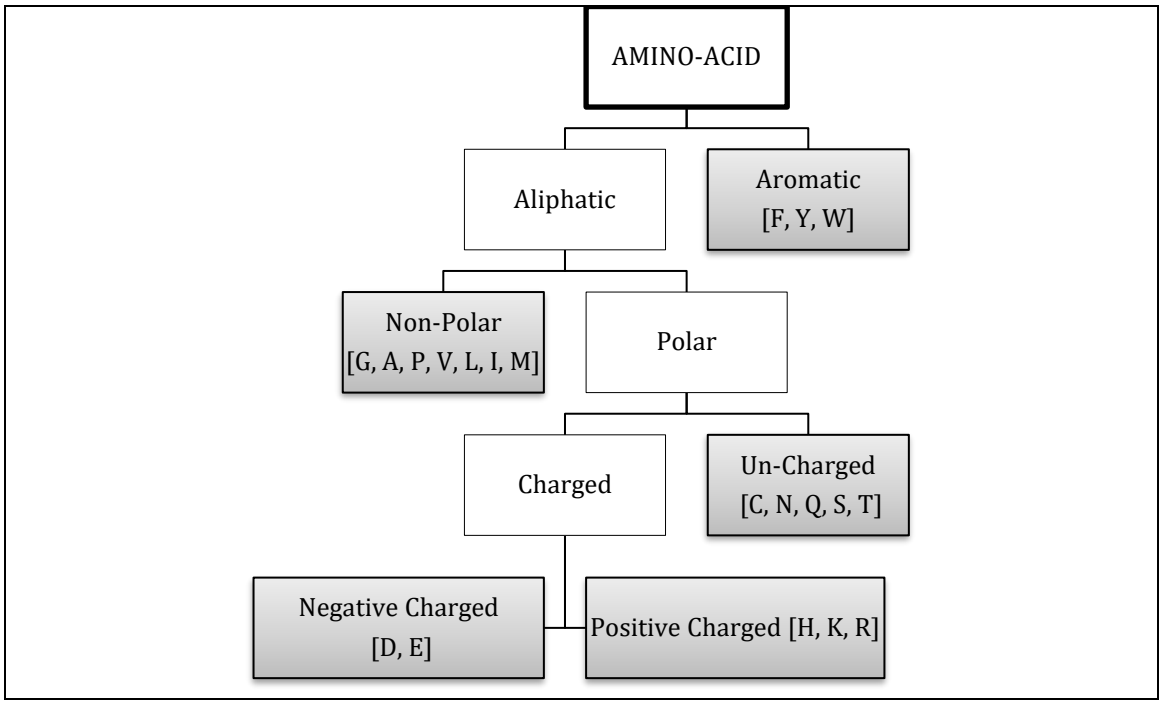

Fig. 3. Amino acids clustering.

$$p1 = (PUAA)_i = \frac{\sum_{i=1}^{7} n_i}{L} \quad \textbf{(3)}$$

Where $(NPUAA)_i$ is the total percentage of Non- Polar uncharged amino acids type and n_i is the counting number of [G, A, P, V, L, I, M] occurring in a protein with sequence length L.

$$p2 = (NPUAA)_i = \frac{\sum_{i=1}^{5} n_i}{L} \quad \textbf{(4)}$$

Where $(PUAA)_i$ is a percentage of Polar aliphatic amino acids type and n_i is the counting number of [S, T, C, N, Q] occurring in a protein with sequence length L.

$$p3 = (PCAA)_i = \frac{\sum_{i=1}^{3} n_i}{L} \quad \textbf{(5)}$$

Where $(PCAA)_i$ is the total percentage of positive charged amino acids type and n_i is the number of [K, R, H] occurring in a protein with sequence length L.

$$p4 = (AAA)_i = \frac{\sum_{i=1}^{3} n_i}{L} \quad \textbf{(6)}$$

Where $(AAA)_i$ is the total percentage of Aromatic amino acid type and n_i is the counting of [F, W, Y] occurring in a protein with sequence length L.

$$p5 = (NCAA)_i = \frac{\sum_{i=1}^{2} n_i}{L} \quad \textbf{(7)}$$

Where $(NCAA)_i$ is the percentage of negatively charged amino acid type and n_i is the number of [D, E] occurring in a protein with sequence length L.

$$p6 = (PUAA)_i = \frac{\sum_{i=1}^{2} n_i}{L} \quad \textbf{(8)}$$

Where $(BKR)_i$ is the percentage of basic KR residues amino acids type and n_i is the number of [K, R] type occurring in a protein with sequence length L.

Amino-acids features represented in a six-dimensional vector which: *FAA = [p1, p2, p3, p4, p5, p6]*

Finally, we used the Euclidian distance between S_i, S_j to identify the closest matching based on Amino-acids clustering.

$$EDAA_{(S_i,S_j)} = \sqrt{\begin{matrix}(p1_i - p1_j)^2 + (p2_i - p2_j)^2 + (p3_i - p3_j)^2 + \\ (p4_i - p4_j)^2 + (p5_i - p5_j)^2 + (p6_i - p6_j)^2\end{matrix}} \quad \textbf{(9)}$$

C. Clustering based on secondary structure

Protein structure is very important to understand protein function. Protein structure has three main levels of protein structure: primary, secondary, and tertiary as explained in figure 4. The primary structure is the simplest level of protein structure that is the sequence of amino acids.

For PSS prediction, one of the most widely used tools is the DSSP (Dictionary of Protein Secondary Structure) package [44]. The program gives the predicted secondary structure, h=helix, e=extended or beta-strand, and c=coil; protein structure data can be obtained from protein data bank (PDB). PSS has three structural domains α-helix, β-strand, and coil. GOR software is one of the PSS prediction methods. GOR version IV is used to predict protein secondary structure (https://npsa-prabi.ibcp.fr). For example, in figure 5 to GOR IV [45] software result. Figure 5 represents an example of a Secondary structure clustering of the protein sequence.

The knowledge of protein structure will be increasing the accuracy and reduce the time for searching to produce the alignment result and provide the protein function information.

$$p_{\alpha_i} = \frac{\sum n_\alpha}{L} \quad \textbf{(10)}$$

$$p_{\beta_i} = \frac{\sum n_\beta}{L} \quad \textbf{(11)}$$

$$p_{c_i} = \frac{\sum n_c}{L} \quad \textbf{(12)}$$

Where $n\alpha$ is the *h* counting number, $n\beta$ is the number of e and *nc* total of *c* numbers in a protein sequence with sequence length *L*. Secondary structure features represented in a 3-dimensional vector which: *FSS = [pα, pβ, pc]*.

We use the Euclidean distance for secondary structure between S_i, S_j to identify the closest matching based on secondary structure as follows:

$$EDSS_{(S_i,S_j)} = \sqrt{(\alpha_i - \alpha_j)^2 + (\beta_i - \beta_j)^2 + (c_i - c_j)^2} \quad \textbf{(13)}$$

Which α_i is a percentage of α-helix to S_i, β_i is a percentage of β-Sheet to S_i, C_i is a percentage of the coil in S_i, α_j is a percentage of α-helix to S_j, β_j is a percentage of β-Sheet to S_j and C_j is a percentage of the coil in S_j.

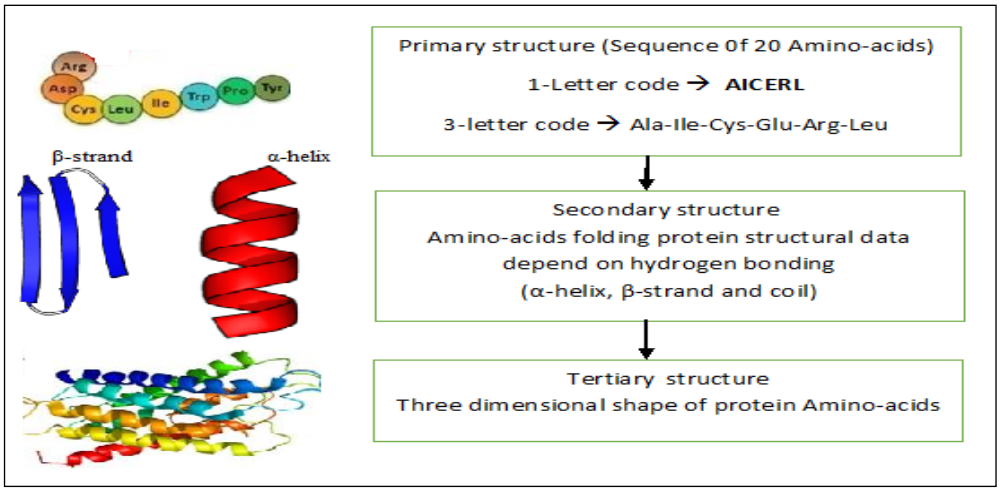

Fig. 4. Protein structure levels.

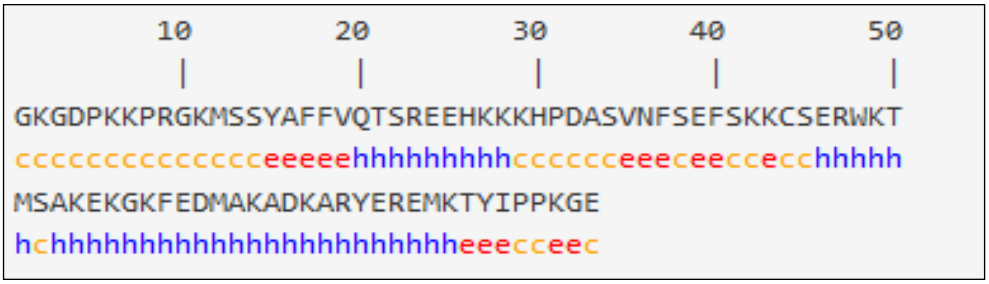

Fig. 5. GOR IV Secondary structure prediction result.

IV. Implementation Environment

Programs were run under the same environment in the cloud platform. The Amazon EC2 cloud platform is used, all programs run on Linux extra-large 4 CPU – 15 GB – 64 bit and Amazon S3 for storage data with cloud Amazon EC2. For performance evaluation, SPscore accuracy, performance measurement is used.

TABLE II. SIX CASES FOR EVALUATION

BALIBASE	Filename	Seq #	Average length	Seq identity
RV11	BB11001	4	86	<20% identity
RV12	BB12043	34	318	20-40% identity
RV20	BB20040	87	482	Up to 3 orphans
RV30	BB30003	142	407	<25% residue identity
RV40	BB40049	62	862	up to 400 residues
RV50	BB50006	60	642	up to 100 residues

SPsore (sum of pairs score) is calculated as the sum of the score for each pair in every column of MSA result and compared with the sum of pairs score for MSA reference. To compute SPscore, we used MSA comparator software MQAT version 2.0.1 [46]. Which it allows comparing between the alignment reference file and more test alignments (>21MB size). MSA comparator is more efficient than a BALIBASE C program [46]. We used six cases for evaluation as shown in table II, the last five cases are the highest sequence number in BAliBASE dataset.

$$SPscore(a_i, \dots, a_j) = \frac{\sum_{i,j}^{n} S(a_i, a_j)}{\sum_{i,j} S_r} \quad \textbf{(14)}$$

Where S_r is the dataset reference score and $S(a_i, a_j)$ score between pairwise sequences a_i and a_j.

V. SIMULATION RESULTS AND DISCUSSION

At first, we evaluate for each feature the accuracy performance. The proposed has six features for Amino-acids, namely *FPUAA*, *FNPUAA*, *FPCAA*, *FAAA*, *FNCAA*, and *FBKR*. It has three features for secondary structure, namely *Fα*, *Fβ*, and *Fc*. Secondly, combine the feature by using the Euclidian distance formula for Amino-acids features, namely *FAA* and combine three secondary structure features namely *FSS*. The proposed has *FLCS* for the longest common subsequence. Finally, combine the three basic features *FLCS*, *FAA*, and *FSS*. After that, apply the PROBCONS MSA tool in parallel for each cluster. To return the final alignment, merge the alignment result for all clusters. The following is the proposed enhancement PROBCONS (E- PROBCONS) algorithm.

E- PROBCONS algorithm
Input: *n* Protein Sequences, $S_1, S_2, ..., S_n$
Output: *n* aligned Protein Sequences, $S'_1, S'_2, ..., S'n$

Key ← *sequence.name[i];*
value ← *sequence [i];*
sequence1name ← *sequence.name[0];*
sequence1value ← *sequence[0];*
Threshold= [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]
For *i*=1 to *n* (each Protein Sequence value *Si*),
  FLCS ← *LCS(sequence1value, Si);*
  FAA ← *AA(sequence1value, Si);*
  FSS ← *SS(sequence1value, Si);*
  x ←*Euclidian_distance(FLCS, FAA, FSS);*
  If (*x* >=*Threshold [j]*)
    Put sequence *key* in file *sfj*
    Put sequence *value* in file *sfj*
End for
For each sequence *i* in file *sfj*
  Apply the PROBCONS MSA tool in MultiProcessing Amazon EC2.
End for
Merge between groups progressively to retrieve MSA.

The clustering stage partitions the large set file into smaller subgroups to reduce time and we can use the PROBCONS tool. PROBCONS has a big problem since the maximum sequence number is limited to 1000 protein sequences. The proposed solved PROBCONS limitation problem. It partitioned the large file into smaller files. The clusters of this file based on some features.

The SPscore for measurement accuracy of all different features appears in Table III. In Table III, we ranked the set of features based on average SPscore for six test cases. The results presented that Amino-acid clustering is affected the accuracy results than without clustering. The LCS feature increasing the accuracy with the PROBCONS tool for MSA. The results show that the combination of the set of listed features is affected the quality of final result alignment.

TABLE III. THE SPSCORE FOR ACCURACY BASED ON DIFFERENT FEATURE CLUSTERING

Clustering	Alignment	BB11001	BB12043	BB20040	BB30003	BB40049	BB50006	Rank
-----	KALIGN	0.482	0.965	2.226	0.658	0.478	0.747	----
----	PROBCONS	0.542	1.063	2.362	0.848	0.656	0.903	----
FLCS	PROBCONS	**0.723**	**1.082**	**2.368**	**0.888**	**0.672**	**0.931**	**11**
FAAA	PROBCONS	1.084	1.072	2.332	0.97	0.707	0.902	6
FBKR	**PROBCONS**	**1.084**	**1.138**	**2.373**	**0.942**	**0.735**	**0.892**	**2**
FNPUAA	PROBCONS	1.084	1.087	2.356	0.961	0.67	0.942	4
FNCAA	PROBCONS	0.723	1.103	2.376	0.968	0.74	0.886	8
FPCAA	PROBCONS	0.723	1.103	2.356	0.93	0.719	0.894	10
FPUAA	PROBCONS	1.084	1.101	2.353	0.965	0.655	0.924	5
Fα	PROBCONS	**0.723**	**1.132**	**2.374**	**1.036**	**1.668**	**0.903**	**1**
Fβ	PROBCONS	0.723	1.205	2.359	0.93	0.672	0.903	7
Fc	PROBCONS	0.723	1.063	2.374	0.942	0.668	0.879	9
FAA	PROBCONS	**1.084**	**1.142**	**2.366**	**0.955**	**0.665**	**0.89**	**3**
FSS	PROBCONS	0.542	1.127	2.359	0.94	0.724	0.87	12
FSS + FLCS	PROBCONS	0.542	1.08	2.36	0.95	0.709	0.851	---
FSS + FAA	PROBCONS	0.723	1.176	2.359	0.848	0.722	0.933	---
FAA + FLCS	PROBCONS	0.542	1.141	2.371	1.036	0.898	0.918	---
FAA + FLCS + FSS	PROBCONS	**1.446**	**1.083**	**2.355**	**1.036**	**0.677**	**0.903**	**---**

TABLE IV. Average execution time for (clustering based on the combination of FAA, FLCS and FSS, alignment, and merge steps) of PROBCONS (with clustering or not) and KALIGN (without clustering).

HOMFAM benchmark	**No- Parallelism**		**4 CPU**
	No- Clustering		**With- Clustering (FAA+ FLCS +FSS)**
Sequence Number	**KALIGN**	**PROBCONS**	**Proposed on Amazon EC2**
100	2500	55110	425
200	7400	234000	772
500	48400	2404000	1695
1000	152500	6663000	3600
2000	663500	…	9174
5000	3384900	...	31264

As shown in table III, figure 6, and figure 7, Fα achieves the highest accuracy, followed by the combination of all features. Finally, for execution time evaluation, we compare between PROBCONS and KALIGN without clustering, FLCS with PROBCONS, FSS with PROBCONS, FAA with PROBCONS, and the combination between FAA, FSS, and FLCS with PROBCONS as shown in figure 7 and Table IV. In table IV, the maximum sequence number is limited to 1000 protein sequences for PROBCONS. The proposed algorithm achieved the highest alignment accuracy. Feature clustering understands protein sequence, structure, and function and all these features affect accuracy strongly and reduce the running time of searching to produce the final alignment result.

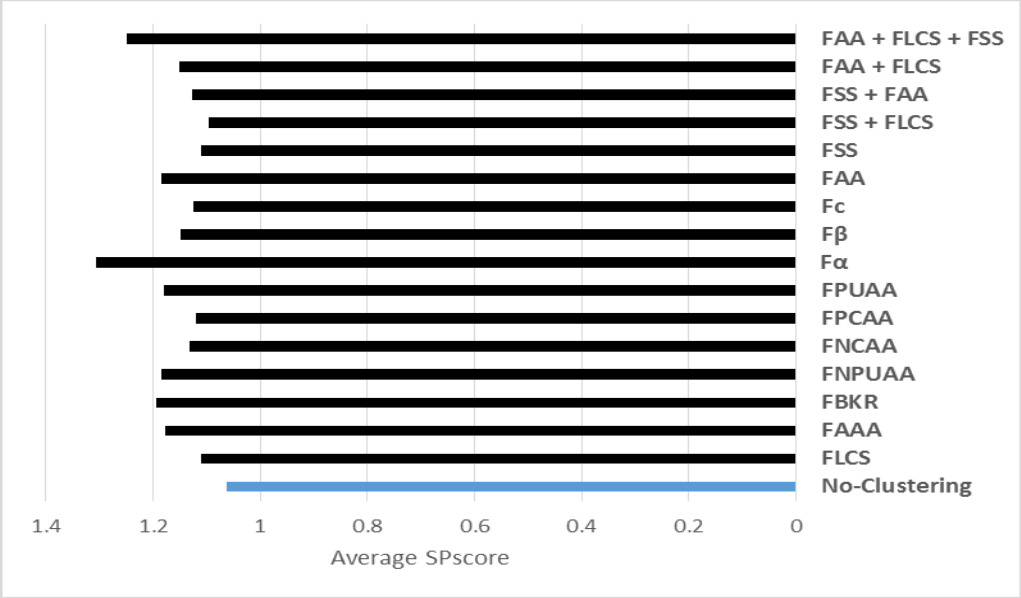

Fig. 6. Average SPscore result for PROBCONS with clustering or without.

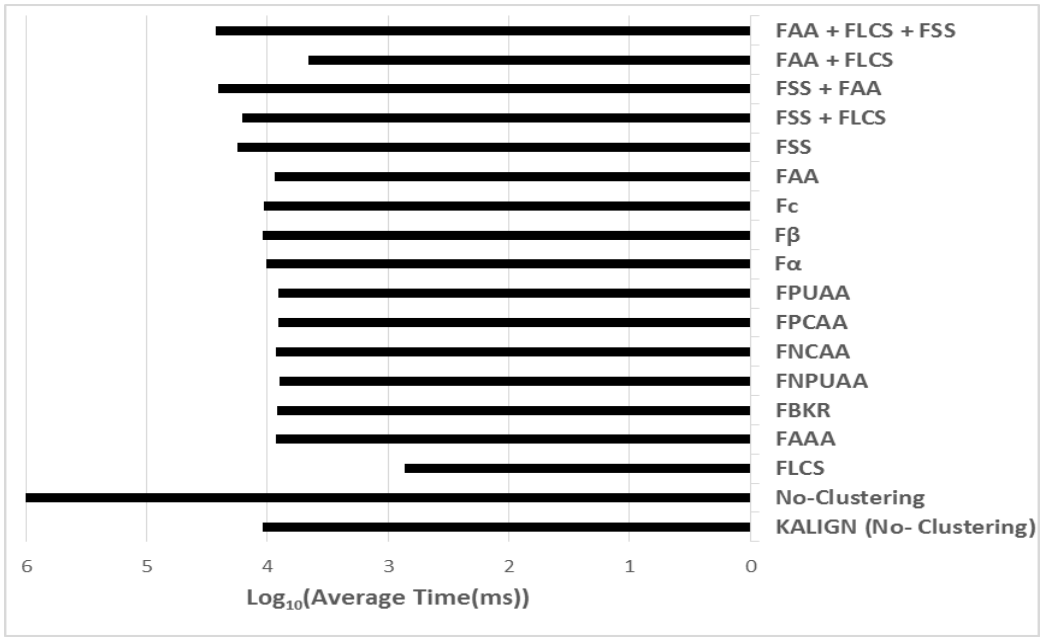

Fig. 7. Average execution time for (clustering, alignment and merge steps) of PROBCONS (with clustering or not) and KALIGN (without clustering).

Figure 8 shows a drastic reduction of runtime in Cloud implementation of proposed progressive MSA with feature clustering, unlike the standalone implementation without clustering. It roughly estimates that the runtime is reduced. For example for 5000 sequences, the proposed progressive MSA take 56.415 minutes (less than one hour) without clustering process, 0.53 minute (less than one minute) with clustering process in Amazon Elastic Compute Cloud (EC2) platform, 0.22 minute (less than half of the minute) with clustering process in Amazon Elastic Map-Reduce (EMR).

The proposed calculation accomplished the most noteworthy alignment precision. Highlight clustering comprehends protein sequence, structure, and function. Finally, every one of these highlights influences exactness emphatically and decreases the running time of seeking to deliver the last alignment result.

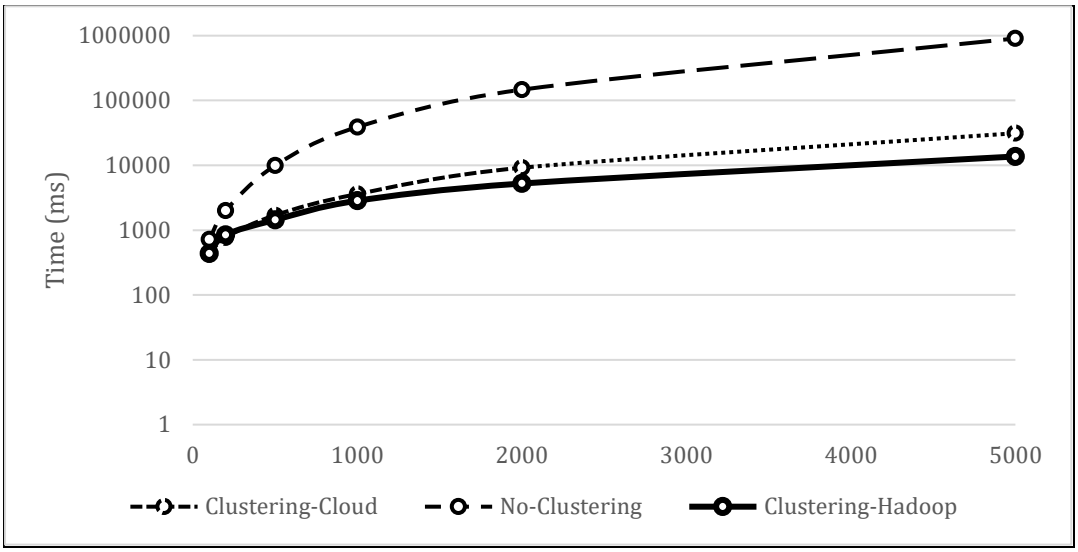

Fig. 8. The time effect of protein feature clustering on cloud or Hadoop.

VI. CONCLUSION

In this paper, clustering the large-scale protein sequences dependent on some biology protein features. To achieve accurate alignment, we classify protein large sequences depend on the LCS percentage, PSS, and amino acid (AA) clustering. PROBCONS tool achieves the highest accuracy but is moderately slow with execution processing. PROBCONS is enhanced by using the biology feature for Clustering on the Amazon EC2 cloud platform. The Cloud platform is used to decline MSA execution time for the PROBCONS tool. The maximum accuracy is achieved based on the combination of the protein biological features and clustering of the large-scale multiple protein sequences. Feature clustering understands protein sequence, structure, and function. All these features affect accuracy strongly and reduce the running time of searching to produce the final alignment result.

REFERENCES

[1] Do CB, Katoh K, " *Protein multiple sequence alignment methods*", Mol Biol Clifton NJ2008, Vol. 484, pp. 379–413, 2008.

[2] M. a. Aniba, " *Issues in bioinformatics benchmarking: the case study of multiple sequence alignment*", Nucleic Acids Res, Vol. 38, pp. 7353–7363, 2010.

[3] Wallace IM, Blackshields G, Higgins DG., "*Multiple sequence alignments*", Current Opinion in Structural Biology, Vol. 15, no. 3, pp. 261-266, 2005.

[4] S. B. Needleman and C. D. Wunsch.*"A general method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins*", Journal of Molecular Biology, Vol. 48(3), pp. 443-453, 1970.

[5] Peng Zhao, Tao Jiang." *A heuristic algorithm for multiple sequence alignment based on blocks*", Combinatorial Optimization, Vol. 5(1), pp. 95–115, Mar 2001.

[6] Feng DF, Doolittle RF., "*Progressive sequence alignment as a prerequisite to correct phylogenetic trees*", Journal of Molecular Evolution, Vol. 4(25), pp. 351-360, 1987.

[7] P.Zhao and Tao Jiang J, Hirosawa, M., Totoki, Y., Hoshida, M. and Ishikawa, M., "*Comprehensive study on iterative algorithms of multiple sequence alignment*", CABIOS, Vol. 11, pp. 13–18, 1995.

[8] Chuong B. Do, Mahathi S.P. Mahabhashyam, Michael Brudno, and Serafim Batzoglou, "*ProbCons: probabilistic consistency-based multiple sequence alignment*", Genome Research, Vol. 2(15), pp. 330-340, 2005.

[9] Stoye J, "*Multiple sequence alignment with the divide-and-conquer method*", Gene 211, pp. GC45–GC56, 1998.

[10] Stoye J, Moulton V, Dress AW, " *DCA: an efficient implementation of the divide-and-conquer approach to simultaneous multiple sequence alignment*", Comput. Appl. Biosci. Vol.13 (6), pp. 625-626, 1997.

[11] Larkin MA, Blackshields G, Brown NP, Chenna R, McGettigan PA, McWilliam H, Valentin F, Wallace IM, Wilm A, Lopez R, Thompson JD, Gibson TJ, Higgins DG. "*Clustal W and Clustal X version 2.0*.", Bioinformatics, Vol. 23, pp. 2947-2948, 2007.

[12] Sievers F, Higgins DG, "*Clustal Omega for making accurate alignments of many protein sciences*". Protein Sci. , Vol. 27, pp. 135-145, 2018.

[13] Lassmann T, Sonnhammer EL., "*Kalign—an accurate and fast multiple sequence alignment algorithm*", BMC Bioinformatics, Vol. 6, pp. 298, 2005.

[14] Lassmann T, Frings O, Sonnhammer EL." *Kalign2: high-performance multiple alignments of protein and nucleotide sequences allowing external features*". Nucleic Acids Res, Vol.37, pp. 858–865, 2009.

[15] Katoh K, Standley DM, "*MAFFT multiple sequence alignment software version 7: improvements in performance and usability",* Molecular Biology and Evolution, Vol. 4(30), pp. 772–780, 2013

[16] Edgar RC, "*MUSCLE: a multiple sequence alignment methods with reduced time and space complexity*", BMC Bioinformatics, Vol. 5, pp. 113-131, 2004.

[17] Thompson JD, Koehl P, Ripp R, Poch O., "*BAliBASE 3.0: latest developments of the multiple sequence alignment benchmark*", Proteins, Vol. 1(61), pp. 36-127, 2005.

[18] Jiang, Q., Jin, X., Lee, S.-J., & Yao, S. "*Protein secondary structure prediction: A survey of the state of the art".* Journal of Molecular Graphics and Modelling, Vol. 76, pp. 379–402, 2017.

[19] D.T. Jones, "*Protein secondary structure prediction based on position-specific scoring matrices*", J. Mol. Biol. Vol. 292, pp. 195–202, 1992.

[20] Z.-H., Zhou, M., Luo, X., & Li, S. "*Highly Efficient Framework for Predicting Interactions between Proteins*". IEEE Transactions on Cybernetics, Vol 47(3), pp. 731–743, 2017.

[21] H. Nakashima, K. Nishikawa, and T. Ooi, "*The folding type of a protein are relevant to the amino acid composition*," J. Biochem., Vol. 99(1), pp. 153–162, 1986.

[22] Bergroth, L., Hakonen, H. and Raita, T. "*A Survey of Longest Common Subsequence Algorithms*". SPIRE (IEEE Computer Society), pp. 39–48, 2000.

[23] Daugelaite, J., O' Driscoll, A., & Sleator, R. D. (2013)." *An Overview of Multiple Sequence Alignments and Cloud Computing in Bioinformatics*". ISRN Biomathematics, pp. 1–14, 2013.

[24] Xing, Z., Pei, J., & Keogh, E.*" A brief survey on sequence classification*". ACM SIGKDD Explorations Newsletter, Vol. 12(1), pp. 40, 2010.

[25] Y. Altun, I. Tsochantaridis, and T. Hofmann. "*Hidden Markov support vector machines*". ICML '03, the Twentieth International Conference on Machine Learning, pp. 3 -10, 2003.

[26] Zhu X, Li K, Salah A." *A data parallel strategy for aligning multiple biological sequences on multi-core computers*". Computers in Biology and Mediciene, Vol. 43(4), pp. 350-361, 2013.

[27] Charu Sharma and A.K.Vyas "*Parallel Approaches in Multiple Sequence Alignments*". International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 4(2), pp 264-276, 2014.

[28] Diana H.P.Low, BharadwajVeeravalli, David A.Bader, "*On the Design of High-Performance Algorithms for Aligning Multiple Protein Sequences on Mesh-Based Multiprocessor Architectures*" Journal of Parallel and Distributed Computing, no. 67(9), pp. 1007-1017, 2007.

[29] Chaichoompu K, Kittitornkun S, and Tongsima S. "*MT-ClustalW: multithreading multiple sequence alignment*"; Proceedings 20th IEEE International Parallel & Distributed Processing Symposium. IEEE Computer Society Press; pp. 280, 2006.

[30] Kuo-Bin Li. "*ClustalW-MPI: ClustalW analysis using distributed and parallel computing". Bioinformatics*. ; Vol.19, pp.1585–1586, 2003.

[31] Quan Zou, Qinghua Hu, Maozu Guo, Guohua Wang. "*HAlign: Fast Multiple Similar DNA/RNA Sequence Alignment Based on the Centre Star Strategy*". Bioinformatics, Vol. 31(15), pp. 2475-2481, 2015.

[32] Shixiang Wan, Quan Zou. "*HAlign-II: efficient ultra-large multiple sequence alignment and phylogenetic tree reconstruction with distributed and parallel computing*". Algorithms for Molecular Biology, pp. 12-25, 2017.

[33] Blazewicz, J., Frohmberg, W., Kierzynka, M., Wojciechowski, P. "*G-MSA - A GPU-based, fast and accurate algorithm for multiple sequence alignment* ". Journal of Parallel and Distributed Computing, Vol. 73(1), pp. 32–41, 2013.

[34] Xi Chen, Chen Wang, Shanjiang Tang, Ce Yu, Quan Zou. "C*MSA: A heterogeneous CPU/GPU computing system for multiple similar RNA/DNA sequence alignment*". BMC Bioinformatics, Vol. 18, pp. 315, 2017.

[35] Deorowicz S, Debudaj-Grabysz A, Gudyś A. "*FAMSA: Fast and accurate multiple sequence alignment of huge protein families*". Sci Rep; Vol. 6, pp. 33964 , 2016.

[36] Deng, Xin and Jianlin Cheng. "*MSACompro: protein multiple sequence alignment using predicted secondary structure, solvent accessibility, and residue-residue contacts*." BMC Bioinformatics, Vol. 12, pp. 472. 2011.

[37] Yongchao Liu and Bertil Schmidt:" *Multiple protein sequence alignment with MSAProbs*". Methods in Molecular Biology, Vol. 1079, pp. 211-218, 2014

[38] Yongchao Liu, Bertil Schmidt, Douglas L. Maskell: "*MSAProbs: multiple sequence alignment based on pair hidden Markov models and partition function posterior probabilities*". Bioinformatics, Vol. 26(16), pp. 1958-1964, 2010

[39] Kleinjung J, Douglas N, Heringa J. "*Parallelized multiple alignments*". Bioinformatics, Vol.18, pp. 1270–127, 2002.

[40] Eman M. Mohamed, Hamdy M. Mousa, Arabi E. Keshk, "*comparative analysis of multiple sequence alignment tools*", MECS, Vol. 10(8), pp. 24-30, 2018.

[41] Paolo Di Tommaso, Miquel Orobitg, Fernando Guirado, Fernado Cores, Toni Espinosa, Cedric Notredame, " Cloud-Coffee: implementation of a parallel consistency-based multiple alignment algorithm in the T-Coffee package and its benchmarking on the Amazon Elastic-Cloud.," Bioinformatics, Vol. 15(26), pp. 1903-1904, 2010.

[42] S.P. Mielke, V.V. Krishnan, "*Protein structural class identification directly from NMR spectra using averaged chemical shifts*", Bioinformatics. Vol. 19, pp. 2054–2064, 2003.

[43] J. Kähärä and H. Lähdesmäki, "*Evaluating a linear k-mer model for protein–DNA interactions using high-throughput SELEX data*," BMC Bioinformat., vol. 14(10), pp. S2, 2013.

[44] Kabsch W, Sander C. "*A dictionary of secondary structure*." Biopolymers; Vol. 22, pp. 2577–2637, 1983.

[45] Sen TZ, Jernigan RL, Garnier J, Kloczkowski A. "*GOR V server for protein secondary structure prediction*". Bioinformatics. Vol. 21(11), pp. 2787–2788, 2005.

[46] Pervez MT, Babar ME, Nadeem A, et al. "*IVisTMSA: Interactive Visual Tools for Multiple Sequence Alignments*". Evol Bioinform Online. Vol. 11, pp.35–42, 2015.