Kashf Journal of Multidisciplinary Research
Vol: 01 Issue 11 (2024)
P-ISSN: 3007-1992 E-ISSN: 3007-200X
https://kjmr.com.pk

GAMSA-ALIGN: A NOVEL APPROACH FOR EFFICIENT PROTEIN SEQUENCE ALIGNMENT USING GENETIC ALGORITHM

MUBASHIR IMAM

Department of Computer Science, Govt College University, Faisalabad, PO 3800 Pakistan

MUEED AHMED MIRZA*

Faculty of Computer Science, Riphah International University, Islamabad, PO 4400 Pakistan

WASIF ALI

Faculty of Computer Science, Capital University of Science and Technology, Islamabad, PO 4400 Pakistan

HASEEB TASLEEM

Faculty of Computer Science, Riphah International University, Islamabad, PO 4400 Pakistan

MUHAMMAD USMAN KARIM

Faculty of Computing, Riphah International University, Islamabad, Pakistan

***Corresponding author: MUEED AHMED MIRZA** (mueedahmed92@gmail.com)*

Article Info

Abstract

Bioinformatics has played an important role in discovering medicine because whole-genome sequences can help to find out the many genetic diseases. Bioinformatics uses computational tools to manage, store and analyze the data. Multiple sequence alignment (MSA) seems to be a very useful process in molecular and evolutionary biology. There are a variety of software's and methods for it. It's used to find conserved patterns, identify protein domains, identify 2D and 3D structures using homology, and conduct evolutionary research. There are several methods for aligning multiple sequences. Many strategies are designed to enhance speed while ignoring the quality of the resulting alignment. Similarly, several strategies are designed to enhance accuracy while ignoring speed. As a result, finding the best method for alignment accuracy and computing cost has become an important factor in choosing the best MSA method. Genetic Algorithms GAMSA-Align have also shown promise in optimizing the multiple sequence alignment process, offering potential improvements in both speed and accuracy. In this study assessed the cost and accuracy of nine common MSA methods against the Bali BASE v4.0 benchmark alignment datasets, including ProbCons 1.12, T-Coffee 9.03, MAFFT 7.031, MUSCLE 3.8.31, Clustal1.1.0, Probalign 1.4, and ProDa, Kalign, and Prank. The two standard scoring procedures, TC score and SP score, are used to calculate alignment accuracy, and computing costs were evaluated by measuring peak memory consumption and CPU execution time. The results indicated that the ProbCons and ProbAlign MSA methods that are based on the progressive consistency approach were first and second, but these tools had a high execution cost.

Keywords: *Multiple Sequence Alignment; Genetic Algorithm; Computational Cost; Bali BASE v4.0; Protein Alignment accuracy*

Introduction

The molecular structure of life is really a complicated system that began roughly 1 trillion years just after start of the universe, around 4 billion years ago. The structure of an organism is stored in DNA, that is then passed to RNA that examines, utilizes, and decodes that instruction to produce a protein. There are hundreds of distinct proteins within humans, one with its own function and structure. Proteins, by reality, are most structurally complex molecules yet discovered. [1] Proteins contain massive molecules made up of smaller repeating molecules that are arranged in a certain order. Polypeptide chains have been composed of amino acids which are linked together by polypeptide chain that create polymers. Several polypeptide chains can make up a protein. A backbone of a protein structure is formed when all amino acids are bound to form residues. [2]

Multiple sequence alignment (MSA) for related proteins is one of the most important challenges in bioinformatics, because its solution may assist predict function of proteins, as well as tell researchers about species evolutionary relationships. Despite major advancements in alignment algorithm performance, obtaining consistently precise alignments may be difficult. [3]

Sequence alignment can be accomplished using one of two ways. The pairwise sequence alignment (PSA) technique as well as multiple sequence alignment (MSA) approach are the two methods. PSA is a process for determining the evolutionary connection between two Protein and RNA/DNA, or sequences by determining the sequences' greatest similarity. MSA stands for Multiple Sequence Alignment, and it is a well-known approach for aligning more than two biological sequences. Because multiple sequence alignment can be such a complicated task, proper MSA can only be computed for a small number of sequences, which is problematic in real-world scenarios. When executing MSA, dynamic programming, as utilized in the paired sequence technique, is challenging for a high number of sequences, hence heuristic methods using estimated approaches have found to be more successful. [4] MSA methods are commonly used to identify a novel protein's family, predict protein structure, and perform phylogenetic analysis. The MSA is based on the idea that all matched biological sequences have evolutionary relationships. In such analysis, the accuracy of alignment findings is critical, but it is clear that the results provided by various alignment methods are rather different. Many MSA methods have been created over time, including as MAFFT, PAGAN, MUSCLE, T-Coffee, PRRP, Kalign, GLprobs, Probalign, NAST, Clustal Omega, Probcons, ProDa, and PRANK etc. [5]

Several researches have already been conducted on the evaluation of MSA algorithms using the benchmark datasets Bali BASE, PDB, SCOP, Pfams and HOMSTRAD. These experiments revealed that neither of the available MSA methods were effective for all types of datasets. Bali BASE was the first benchmark alignment database made entirely to test the correctness of the MSA method. Equidistant sequences make up Reference 1. Based on the basic similarity of the data set. Protein groups having orphan sequences are represented in Reference 2. Reference 3 refers to a group of subfamilies that are distinct from one another. Lower DNA sequence identity is a characteristic of subfamilies that consists of groups less than 20%. A prototypical RefSeq product that possesses big N/C terminal ends is called Reference 4. The data contain for both big indels and inseres corresponds to the References number 5. RDS 6 constitutes an intricate Description Language that consists of repetitions. [6].

The problem of MSA is caused by the diverse lengths of sequence patterns, inability to unambiguously designate bases because of ambiguity, and the presence of gaps or mismatches. Unlike traditional methods in which spacecraft must be sent point-to-point or navigated along a predetermined trajectory, GAs function by representing potential alignments as sequences of characters (chromosomes) and applying genetic operators like mutation, crossover, and selection to generate a new set of alignments. Therefore, this approach helps the GAs to run an exhaustive search and infuse the domain-wise expertise into the optimization process. Drawing evidence from the evolutionary idea, the GAs was proven to be powerful for MSA algorithms accuracy and efficiency, with them becoming an ideal tool for sequential alignments that is equipped with a strong theoretical basis. [7]

The genetical algorithms (GAs) turned out to be one of the most powerful solutions for the bioinformatics issues involving complex combinatorial problems like DNA, RNA, and protein sequence alignments. They are heavily built on the principles of Darwinian evolution and genetic diversity that make them very good at finding the best / or nearly the best solutions in the vast and very complicated search-spaces and thus they are the optimum tool for dealing with multiple sequence alignment (MSA). Multiple sequence alignment is one of the most important tasks in molecular biology and its application for revealing the similarities and divergence of a given set of data is vital. The challenging part of this work is since the search space is vast thus processing of large datasets require a high computational speed [8]. Conventional methods (including implementations of algorithms that are both accurate and computationally efficient) bring up the issue of novel techniques like GAMSA, implying that such innovative approaches are a matter of critical importance for the development of accurate and computationally efficient algorithms.

II. RELATED WORKS

The passage has the significant information about the multiple sequence alignment algorithm's (MSA) work as a bioinformatic tool which began with basic genetic algorithms and eventually led to key advancements including the MSA method. We highlight the welcome development of sophisticated methods for handling the big biological data as an example of emerging intelligent tools. Additionally, we focus on the significance of key databases, BALIBASE 4 and review various MSA tools, emphasizing their contribution to advancing MSA methodologies and the genetic algorithms' potential in this evolving field.

A.MULTIPLE SEQUENCE ALIGNMENTS METHODS

Researcher compare the ten most popular (MSA) tools, namely, ProbCons, Dialign-TX, MAFFT, MUSCLE, SATe, Kalign, T-Coffee, Multalin, MAFFT (L-INS-i), and Clustal Omega is presented. Researchers also identified importance of different implementations within each tool's algorithms. Using R simulated trees with varying numbers of species,400 actual alignments, and indel Seq-uence Generated was used to create sequence file**s. For investigate the effects of sequence leng**th, exon / intron size, and deletion, a total number 4000 testing alignment was created. Deletion rate. Researchers used Tukey post hoc analysis to produce Multiple Comparisons Table (MCT), which corroborated our findings with, MAFFT(L-INS-i) SATe, or ProbCons have been the most effective methods. The impact on insertion rate overall alignment performance was also investigated, SATe beat most other MSA techniques in terms of alignment accuracy as evaluated by SPS and CS. T-Coffee had the lowest CS and SPS for both biological metrics, insertion rate and deletion rate. [9]

The authors say that MSAcomp, MSAgen, FASTA generator, MSApad, Distance Matrix calculator, IDMC, and Tree calculator are part of the IVisTMSA software product, which includes seven graphical users' interface tools MSAcomp is a program that compares multiple MSAs at once. MSAcomp is really a powerful tool that calculates the CS and SPS over 11298 sequences within just 12 seconds. To create the right test alignment, all chosen MSA techniques were given a sequences file generated by iSG. CS and SPS were used to compare/evaluate the testing alignment and the reference alignment. [10]

The most frequently use multiple alignment benchmarks, Bali BASE, has released a new version that delivers high-quality, carefully polished reference alignments constructed on 3D structural super locations. Bali BASE 3.0 now contains additional, more difficult test cases that reflect the real-world challenges of aligning huge collections of complicated sequences. The volume of protein sequences in the benchmarking has been expanded using a unique, automated updating procedure, and representative testing requirements that represent most of the protein fold space are now accessible. Bali Base's overall number protein proteins also has risen substantially, from 1444 to 6255 sequences. In addition, with all test cases that are tough both for global and local alignment algorithms, full-length sequences now are given [11]. DIALIGN is a powerful multiple sequence alignment tool that excels at detecting local homologies between sequences. A protein alignment generator, which compares inputs to a Pfams database about protein domains, is the most notable feature. The sparse dynamic programming' method is employed in DIALIGN 2.2 to get an optimum alignment mostly in sense of a sequence 'optimization issue. DIALIGN TX's greedy algorithm often chooses random similarities among input sequences for sequence homology. The first step in this technique is to compare sequence data to Pfams have used 'HMMER,' a software that assigns quality marks for match among query protein sequences and protein domain database models. For E-values of such results, we employ a variety of threshold values. [12]

Although the number of MSA techniques available has grown in recent decades, they may be divided into three categories: progressive-based methods (such as CLUSTALW, MAFFT, and MUSCLE), T-Coffee, ProbCons, and certain versions of MAFFT that can be packed and run locally. Different MSA methods might result in different alignments, which will clearly affect all subsequent studies. MSA is usually straightforward because the SARS-CoV-2 genomes are so comparable in sequence, with minimal insertion–deletion events (indels). [13] The starting MSAs are picked from the outputs of M-Coffee and ProbCons, two prominent protein sequence alignment tools. After numerous rounds of the program, the authors used a genetic method's iterative algorithm to find a best protein alignment. As a result, they created the Protein Alignment by Stochastic Algorithm, a novel MSA computational tool (PASA). The efficiency of protein alignments has been measured using the Total Column Score (TC). The PASA technique is put to test mostly on famous Bali-base version 3 benchmarks. The QSCORE software is used to calculate the results. In respect of Q score, the PASA surpasses the MCoffee by 0.7 percent, 14 percent over the ClustalW, and 9.28 percent over MAFFT, 1.2 percent over the ProbCons, primarily on Bali-BASE 3 benchmarks. [14]

B. GENETIC ALGORITHMS

GAs operate on encoded versions of problem parameters, using a population of potential solutions and probabilistic rules to guide their search. The core principles of GAs include fitness-based selection, mating operators for generating offspring, and genetic operators to mutate and crossover genetic material. This process enables GAs to adapt over generations, combining and mutating candidate solutions to explore and exploit the search space efficiently. Particularly effective for tackling complex problems with large and ambiguous search areas, GAs' ability to learn from previous searches allows for progressively better solutions, making them ideal for complex tasks like multiple sequence alignment [15]. Traditional Genetic Algorithms (GAs) are prone to getting trapped in local optima, especially in large-scale problems with sparse distributions of excellent individuals. This limitation hinders the search for global optima and reduces the algorithm's efficiency. To address this, the Multiple Optimal Solutions Genetic Algorithm (MOSGA) is introduced, which aims to generate multiple unique optimal solutions in a single solving process. The research puts forward the flexibility taking lead in the priority sequence planning looks towards handling the uncertainties of part capacity, delivery streamlines, and tool availability which may make the production process less efficient in terms of resource utilization. [16] Researcher developed a new component called assembly sequence planning (ASP) which is dedicated for manufacturing of discrete products that help in deciding the correct order of assemblage. Their methodology employed a non-dominated sorting genetic algorithm alongside a mixed chromosome coding technique for devising solutions. Nonetheless, the approach's focus on a restricted set of constraint variables led to subpar convergence outcomes, diminishing its effectiveness in practical machining contexts. [17] In bioinformatics, simplifying multiple sequence alignment (MSA) into linear models may compromise alignment accuracy and gap penalty assessments. Given MSAs' classification as NP-complete problems, genetic algorithms (GAs) and variations like Non-dominated Sorting Genetic Algorithm-II (NSGA-II) offer innovative solutions for optimizing alignments. This review suggests using GAs and NSGA-II for MSA, treating it as a multi-objective optimization issue, and highlights the significance of tailored GAs and precise mathematical formulations to improve MSA optimization effectively. [18]

Our methodology resonates with the PASA framework, adopting a real MSA representation to ensure accuracy and reliability. We benchmarked against the Bali BASE v4.0 databases, utilizing TC and SP scores to evaluate the performance of nine MSA methods in terms of accuracy, speed, and memory usage. This analysis aims to guide non-specialist biologists in selecting appropriate MSA techniques, with a focus on accuracy and computational efficiency. The latest Bali BASE update, incorporating linear motifs, broadens our analysis scope. The computational methodology section outlines our chosen strategies and implementation, explaining the logic of each step. The results section presents findings from applying these methods to the Bali BASE datasets, offering a thorough evaluation of each MSA technique's effectiveness. The sequence alignment task in GAMSA is proposed through its mechanisms of representation, selection, evaluation, crossover, and mutation. The paper concludes with a detailed bibliography, anchoring our study within the broader scientific literature.

III. COMPUTATIONAL METHODOLOGY

This section covers the dataset selection and experimental setup, detailing the Genetic algorithm's adaptation to solve the Multiple Sequence Alignment (MSA) problem. We explore nine MSA tools, each utilizing distinct methodologies. Understanding these tools and their approaches is crucial for selecting the most effective solution. This comprehensive overview lays the foundation for evaluating their performance in MSA tasks.

A. DATA SELECTION

The study utilized the Bali BASE v4.0 dataset, available at http://www.lbgi.fr/balibase/, which provided six datasets in. MSF or. TFA format, comprising a total of 386 testing sequences along with their respective reference alignments. MSA Methods. We utilized Clustal Omega, T-Coffee, MAFFT, ProbCons, MUSCLE, PRANK, ProDa, Proalign, and Kalign to analyze the 386 query sequence sets obtained in step 1. Default parameters and protein alignment were used for all methods. Each method produced 386 (for Bali BASE v4.0), resulting in a total of 3474 (386 * 9) test alignments.

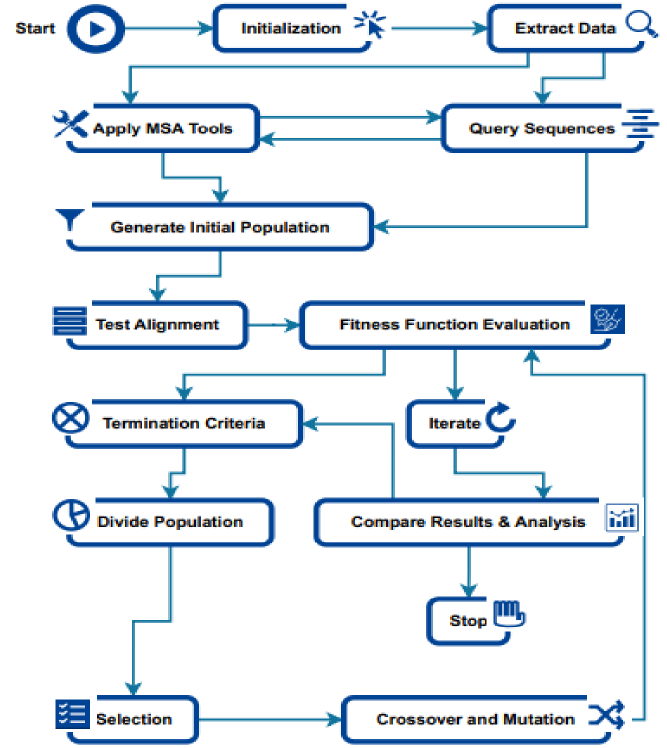


Figure 1: GAMSA flow chart

The execution time of each MSA for all 386 test alignments was recorded. Table 1 provides details of the applications, including their versions and download URLs.

Table 1 Alignments Tools

Software	Versions	Download Link
CLUSTAL O	1.2.4-1	http://www.clustal.org/download/current/
MAFFT	7.310-1	http://mafft.cbrc.jp/alignment/software/
MUSCLE	1:3.8.31	http://www.drive5.com/muscle/downloads.htm
T-Coffee	11.00.8	http://www.tcoffee.org/Projects_home_page/
Probalign	1.4-7	http://probalign.njit.edu/standalone.html
PRANK	1.7.1	http://wasabiapp.org/download/prank/
Probcons	1.12-12	http://probcons.stanford.edu/download.html
Kalign	1:2.03	https://msa.sbc.su.se/cgi-bin/msa.cgi
ProDa	1.0-12	http://proda.stanford.edu/

B. PROPOSED METHODS

There needs to be an adequate summary of references to describe the current state-of-the-art or a summary of the results. Genetic algorithms (GAs) play a crucial role in optimizing protein multiple sequence alignment (MSA). By initializing parameters and extracting data from databases such as BALIBASE 4.0, GAs facilitates the alignment process. Query sequences are prepared for alignment, and each individual in the GA population is assessed using a fitness function. MSA tools are then employed to generate initial alignments, which are subjected to a test alignment to evaluate their quality. The method of the genetic algorithm is detailed in figure # 3. The GA iterates, selecting individuals for reproduction based on their fitness, and calculates scores such as Sum-of-Pairs (SP) and Total Column (TC) scores for each alignment.

$$\mathrm{SP} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} S\,(i,j) \qquad (1)$$

In the Sum-of-Pairs (SP) score equation for multiple sequence alignment (MSA), the index j iterates from i+1 ton, where n represents the total number of sequences in the alignment. The variable S (i, j) denotes the score for aligning residues at positions i and j across all sequences in the alignment. The term Tcc represents in eq # 2, the sum of identical columns Ci out of a total of m identical columns.

$$Tcc = \sum_{i=1}^{m} Ci \qquad (2)$$

The process of mutation involves inserting gaps randomly in each alignment with a fixed probability (p), which is calculated using the formula:

$$P = \frac{\ln{(xy)}}{1-x*10} \qquad (3)$$

where x is the maximum sequence length, y is the number of sequences, and I is the number of columns with identical residues without gaps. We examined numerous alignment datasets, inserting gaps randomly with varying probabilities to enhance alignment scores. Equation 3 was identified to be effective for this purpose, with gaps inserted into the multiple sequence alignment (MSA) during each iteration of the genetic algorithm.

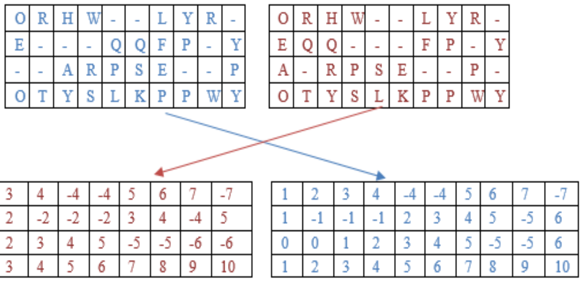
In GAMSA, the affine gap penalty approach using SuiteMSA employs penalties for gap opening and extension. After optimizing penalties in the range of -5 to -20 for gap opening and 0 to -2 for gap extension, optimal values of -15 and -0.9 were found, respectively. This approach excludes terminal gaps from alignment score calculation for improved accuracy. We investigated the incorporation of sequence weights in Bali BASE v 4.0 to address unequal representation on Ref Set #3, but found only marginal improvement in alignment accuracy.

C. ALIGNMENT REPRESENTATION

In GAMSA, the population representation is crucial, impacting algorithm behavior and efficiency. Traditional GA approaches for MSA use binary strings, which increase complexity and space. To improve this, we use integer coding for matrix representations. Residues are coded by sequence positions, and gaps by negative values for the last residue positions. This simplifies crossover positioning and reduces errors, enhancing alignment management in GAMSA.

D. CROSSOVER OPERATOR

During crossover, new individuals (offspring) are created by combining genetic material from two parent individuals. This process involves selecting a crossover point in the parent sequences and swapping the genetic material beyond that point to create the offspring sequences. The crossover point is chosen randomly, and this process helps in exploring new genetic combinations that may lead to better solutions. The NSGA-II algorithm selects the best alignments for the next generation based on their fitness. The process continues until a termination criterion is met, resulting in improved alignments.



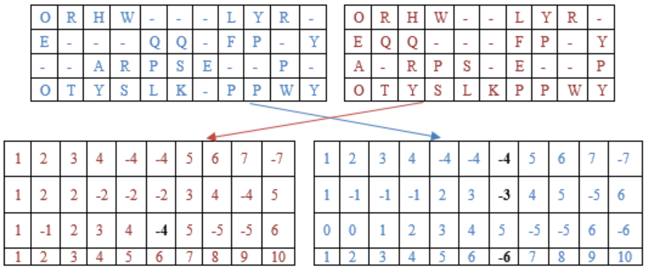
O	R	H	W	-	-	L	Y	R	-
E	-	-	-	Q	Q	F	P	-	Y
-	-	A	R	P	S	E	-	-	P
O	T	Y	S	L	K	P	P	W	Y

O	R	H	W	-	-	L	Y	R	-
E	Q	Q	-	-	-	F	P	-	Y
A	-	R	P	S	E	-	-	P	-
O	T	Y	S	L	K	P	P	W	Y

1	2	3	4	-4	-4	5	6	7	-7
1	2	2	-2	-2	-2	3	4	-4	5
1	-1	2	3	4	5	-5	-5	-6	-6
1	2	3	4	5	6	7	8	9	10

1	2	3	4	-4	-4	5	6	7	-7
1	-1	-1	-1	2	3	4	5	-5	6
0	0	1	2	3	4	5	-5	-5	6
1	2	3	4	5	6	7	8	9	10

Figure 2 Parent 1 & Parent 2 values matrix



O	R	H	W	-	-	-	L	Y	R	-
E	-	-	-	Q	Q	-	F	P	-	Y
-	-	A	R	P	S	E	-	-	P	-
O	T	Y	S	L	K	-	P	P	W	Y

O	R	H	W	-	-	L	Y	R	-
E	Q	Q	-	-	-	F	P	-	Y
A	-	R	P	S	-	E	-	-	P
O	T	Y	S	L	K	P	P	W	Y

1	2	3	4	-4	-4	5	6	7	-7
1	2	2	-2	-2	-2	3	4	-4	5
1	-1	2	3	4	-4	5	-5	-5	6
1	2	3	4	5	6	7	8	9	10

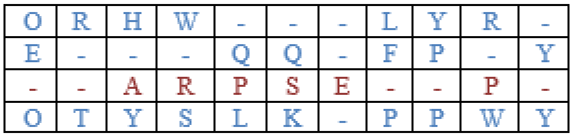
1	2	3	4	-4	-4	-4	5	6	7	-7
1	-1	-1	-1	2	3	-3	4	5	-5	6
0	0	1	2	3	4	5	-5	-5	6	-6
1	2	3	4	5	6	-6	7	8	9	10

Figure 3 Child 1 & Child 2 values matrix

The crossover strategy exchanges genetic information between randomly selected chromosomes Parent 1 and Parent 2 to create new populations (Child 1 and Child 2). Our enhanced one-point crossover considers existing gaps in the alignment. It joins the two parent alignments using a single exchange procedure, combining features of local arrangement mutation and crossover. This operation occurs based on the crossover probability Pc.

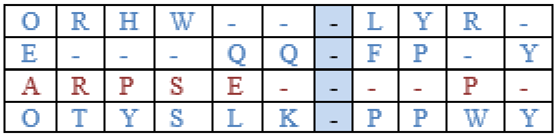
E. MUTATION OPERATOR

The mutation operation in our method maintains population diversity by randomly transmitting genetic information among individuals and recovering missing data. It prevents algorithm entrapment at local minima. We use various mutation operations like gap merging, insertion, single gap, and block gap. After crossover, offspring undergo mutation based on random selection guided by mutation probability. Each mutation operator is applied sequentially to find the one yielding the highest score, with the best chosen and others rejected. If none provide an optimal outcome, a different operator is selected.



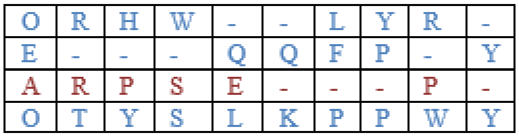
O	R	H	W	-	-	-	L	Y	R	-
E	-	-	-	Q	Q	-	F	P	-	Y
-	-	A	R	P	S	E	-	-	P	-
O	T	Y	S	L	K	-	P	P	W	Y

Figure 4 Child before Mutation



O	R	H	W	-	-	-	L	Y	R	-
E	-	-	-	Q	Q	-	F	P	-	Y
A	R	P	S	E	-	-	-	-	P	-
O	T	Y	S	L	K	-	P	P	W	Y

Figure 5 Shift Blocks of Child Mutation



O	R	H	W	-	-	L	Y	R	-
E	-	-	-	Q	Q	F	P	-	Y
A	R	P	S	E	-	-	-	P	-
O	T	Y	S	L	K	P	P	W	Y

Figure 6 Remove full Colum's of GAPS Mutated Child

To apply mutation in GAMSA, the algorithm targets specific sequences and blocks of gaps. Where the 3rd sequence has a block of gaps that needs to be moved to column 6th. In this case, before the mutation, Child 1 contains a block of gaps that will be shifted to column 6th as part of the mutation process in the Genetic Algorithm for Multiple Sequence Alignment (GAMSA).

F. TERMINATION CRITERIA IN GAMSA

Termination criteria play a vital role in stopping the algorithm when it has achieved a satisfactory solution or when further optimization is not yielding substantial improvements. In our GAMSA approach, we aim to reach the optimal score of the leading chromosome. We employ a termination criterion based on the persistence of fitness scores of the top-performing solutions. Unlike other methods that strive to guarantee the conservation of scores for 100 generations, the proposed method needs to ensure a selection process that remains stable without altering scores of consecutive 100 generations. This not only lowers computational time but also reduces memory usage. An MSA genetic algorithm named GAMSA is developed by the author, which aims to optimize alignments of a specific set of sequences or, in other words, to adjust the alignment parameters or weight coefficients for each sequence in the set to make this alignment best possible. The algorithm takes several parameters: that the first population (P), the crossover likelihood (Pc), mutation possibility (Pm), and maximum generation number (*MaxGen*) of the algorithm must be stated. It is embedded with various functions which perform the following tasks such as initialize population *(IP*), evaluate fitness (*EF*), selection population (*S*), crossover (*C*), mutation (M), next generation selection (*NGS*). These functions examine the fitness function at each iteration by aggressive population until an optimized solution is found.

G. GAMSA ALGORITHM

In the proposed Genetic Algorithm for Multiple Sequence Alignment (GAMSA), the first step involves initializing the algorithm by generating an initial population of alignments.

Algorithm: *Genetic Algorithm for Multiple Sequence Alignment (GAMSA)*

```
Input: $pop_{size}$      ▻ Population size,
       $p_c$             ▻ Crossover probability,
       $p_m$             ▻ Mutation probability,
       $max_{gen}$       ▻ Max number of generations.
Output: - Best alignment found in the final population
$init_{pop}$ ← bestMSAToolsAlignment ($pop_{size}$)▻ initial population
While (maximum of $max_{gen}$ ) do
    $fitness_{vlaue}$ ←spScore ($pop_{size}, init_{pop}$)
    $p_1$ , $p_2$ ← NSGA-II ($fitness_{vlaue}$ , $init_{pop}$)
    $off_{spring}$ ← performCrossover ($p_1$ , $p_2$ , $p_c$ )
    $mutated_{offSpring}$ ← perform Mutation ($off_{spring}, p_m$ )
        ▻ mutation operators (e.g., gap merging, gap insertion, block-gap mutation).
    fitness ← evaluate Fitness ($mutated_{offSpring}$ )
    $p1_{next}$ , $p2_{next}$ ← NSGA-II (fitness)
    update Current population ($p1_{next}$ , $p2_{next}$ )
end while
```

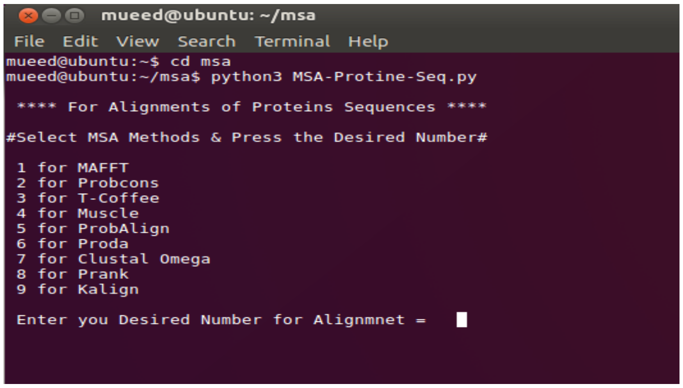
This population, represented as 〖pop〗_size, consists of a specified number of individuals, each of which represents a potential solution to the multiple sequence alignment problem. The size of the initial population is determined by the input parameter 〖pop〗_size, which specifies the number of alignments in the population. The initial population can be created using a method like bestMSAToolsAlignment, which is probably what these functions use nowadays to make alignments with already created multiple sequence alignment tools, producing a set of initial alignments. These original combinations have a vital role in the genetic algorithm's cyclical optimization process that is continuous.

GAMSA (P, Pc, Pm, MaxGen) =NGSA(M(C(S(EF(IP(P))), Pc), *Pm), MaxGen)* (4)

In the main loop of the algorithm, which iterates for a maximum of 〖max〗_gen generations, several key steps are performed. Firstly, the relative fitness or score of population members is assessed using a scoring function, such as the SP score. Then, NSGA II selection for parent allele alignments by their fitness will be done from the population after the running the NSGA-II algorithm. Subsequently, crossover is performed between selected parents with a probability Pc to create offspring alignments. These offspring alignments are then mutated with a probability Pm using mutation operators, such as gap changes. The fitness of the mutated offspring alignments is evaluated, and individuals for the next generation are selected based on their fitness scores using NSGA-II. Finally, the current population is replaced with the selected individuals.

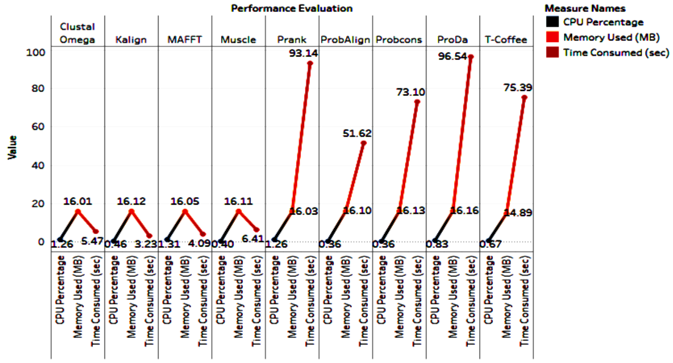
The algorithm continues to iterate through the main loop until the termination criterion is met. The termination criterion could be a maximum number of generations, achieving a satisfactory solution, or a lack of significant improvement in alignment quality. Once the termination criterion is met, the best alignment found in the final population is returned as the output of the algorithm.

IV. SIMULATION AND RESULTS



```
mueed@ubuntu:~$ cd msa
mueed@ubuntu:~/msa$ python3 MSA-Protine-Seq.py

 **** For Alignments of Proteins Sequences ****

#Select MSA Methods & Press the Desired Number#

 1 for MAFFT
 2 for Probcons
 3 for T-Coffee
 4 for Muscle
 5 for ProbAlign
 6 for Proda
 7 for Clustal Omega
 8 for Prank
 9 for Kalign

 Enter you Desired Number for Alignmnet = 
```

Figure. 6 Python Script for calculating the Computation Cost



The trends of CPU Percentage, Memory Used (MB) and Time Consumed (sec) for Performance Evaluation.
Color shows details about CPU Percentage, Memory Used (MB) and Time Consumed (sec).

Figure 7: Figure. 7 Overall Average runtime, memory & CPU performance evaluations of MSA method

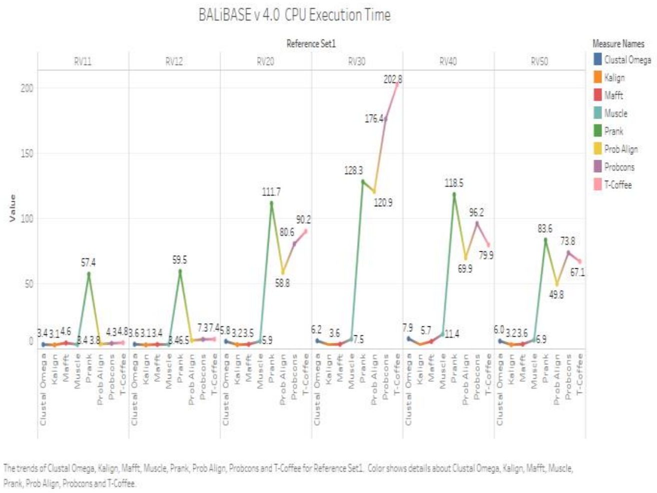
The Genetic Algorithm for Multiple Sequence Alignment (GAMSA) was applied to evaluate nine MSA methods, comparing their performance on Bali BASE v4.0 benchmark datasets. Six methods, including ProbCons 1.12, T-Coffee 9.03, MAFFT 7.031, Clustal1.1.0, Probalign 1.4, and MUSCLE 3.8.31, utilized the progressive approach, while Kalign, ProDa, and Prank employed the non-progressive technique. Evaluation criteria included the TC score and SP score, with computational tests conducted on a system with a Core i5-8110 MHz CPU, 16 GB RAM, and Ubuntu 12.04 OS.

A. PRECISION EVALUATION

A Python script was utilized to calculate execution time, memory usage, and CPU average for each MSA method. Accuracy was assessed using the SP score and TC score, with the SuiteMSA software employed for method comparison [19]. Higher scores indicated better alignment accuracy.

In this research, we conducted an analysis using reference alignments from Bali BASE 4.0 to evaluate the efficiency of the MSA methods mentioned. These benchmarks allowed us to assess the CPU execution time for each method. Additionally, we extended the test cases by introducing new reference sets, totaling 32, with varying MSA challenges. Bali BASE 4.0 comprises 386 reference alignments, encompassing 20,892 protein sequences. We evaluated the efficiency of the protein MSA methods based on CPU time, memory consumption, and CPU percentage. Figure # 7 illustrates the use of CPU time to calculate the total time required to align all sequences with the benchmark. The comparison of execution times

Figure #8.



The trends of Clustal Omega, Kalign, Mafft, Muscle, Prank, Prob Align, Probcons and T-Coffee for Reference Set1. Color shows details about Clustal Omega, Kalign, Mafft, Muscle, Prank, Prob Align, Probcons and T-Coffee.

Figure. 8 Average CPU Time Execution

B. COMPUTATIONAL COST

CPU execution time was analyzed for various methods using reference alignments from Bali BASE 4.0. MAFFT and Kalign demonstrated superior speed, while T-Coffee exhibited the lowest memory consumption. ProDa was the slowest method and utilized the most RAM. In certain cases, ProbAlign and ProbCons exceeded the 5.5-hour execution time threshold.

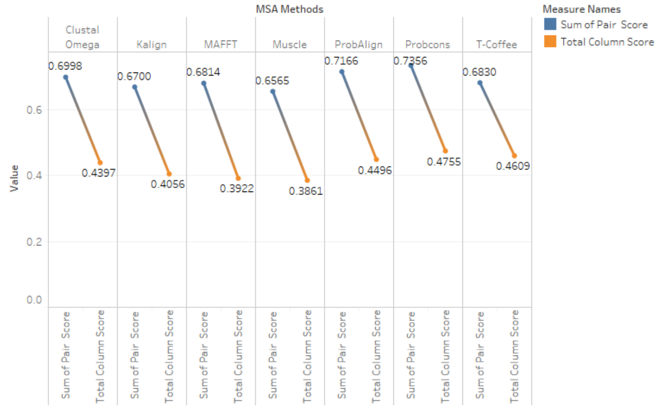
reveals that Kalign is the quickest method, demonstrating a significant advantage over other methods as the number of sequences increases.

In contrast, ProDa is the slowest, taking 10 hours and 35 minutes to align 386 sequences, much longer than Kalign, which only needs 21 minutes. T-Coffee stands out for its efficient memory utilization during sequence alignments, using only 14.89 MB to align 386 sequences compared to ProDa's 16.16 MB. MAFFT and Kalign emerged as the quickest methods in our tests, with T-Coffee exhibiting the smallest memory consumption. MAFFT was notably faster in multi-core mode than Muscle and Clustal Omega, as shown in

However, T-Coffee's drawback is its longer execution time. In the alignment of full-length sequences in five reference sets, ProbAlign and Probcons performed similarly. Probcons, when run in four-core mode, used significantly more RAM than other methods and was generally slower. In the RV30 subset, both ProDa and Prank exceeded the 7.5-hour threshold.

Scoring Methods for MSA Alignments

As opposed to the TC score the SP score acts as a key metric used by the GAMSA to evaluate the quality and accuracy of multiple sequence alignments. These scores are the key in determining the performance of the process of training and encouraging the search of the optimal genetic algorithm. Score of TC is a numerical message that communicates the quantity of conserved and aligned amino acid residues that are similar and identical, they give an idea of the algorithm accuracy and level of alignment quality. This is, however, an additional feature of the SP scoring, which considers the identity of any between alignments of the sequences as a whole, and, thus, is an absolute value. Through this method, GAMSA can more favorably take in these score metrics during the genetic algorithm's fitness evaluation process that will help it to enhance better alignment values as well as the convergence towards the best probable solutions for the multiple sequence alignment problem.

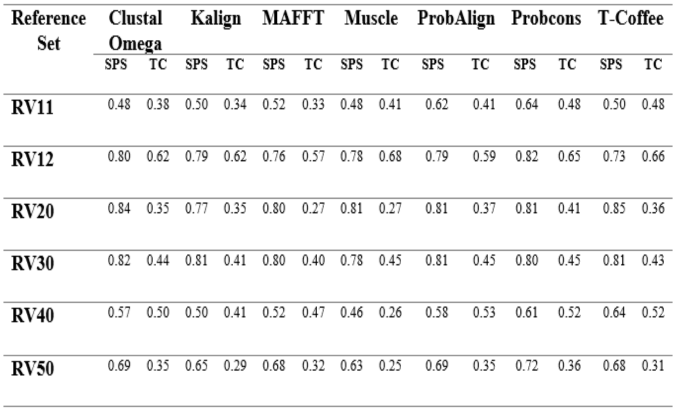


The trends of Sum of Pair Score and Total Column Score for MSA Methods. Color shows details about Sum of Pair Score and Total Column Score.

Figure. 9 SP score and TC score based on the average

In GAMSA (Genetic Algorithm for Multiple Sequence Alignment), the accuracy of MSA (Multiple Sequence Alignment) approaches is determined as per the TC score and SP score. These scores also act as actualizing agents of matching approaches that help in determining the effectiveness of varied MSA techniques. The study focused on the Bali BASE v4.0 benchmark dataset and calculated the mean TC and Scores as defining measure of accuracy for 386 test alignments.

Figure. 10 Average values of the SP Score, TC Score for Bali BASE v4.0



Reference Set	Clustal Omega		Kalign		MAFFT		Muscle		ProbAlign		Probcons		T-Coffee	
	SPS	TC	SPS	TC	SPS	TC	SPS	TC	SPS	TC	SPS	TC	SPS	TC
RV11	0.48	0.38	0.50	0.34	0.52	0.33	0.48	0.41	0.62	0.41	0.64	0.48	0.50	0.48
RV12	0.80	0.62	0.79	0.62	0.76	0.57	0.78	0.68	0.79	0.59	0.82	0.65	0.73	0.66
RV20	0.84	0.35	0.77	0.35	0.80	0.27	0.81	0.27	0.81	0.37	0.81	0.41	0.85	0.36
RV30	0.82	0.44	0.81	0.41	0.80	0.40	0.78	0.45	0.81	0.45	0.80	0.45	0.81	0.43
RV40	0.57	0.50	0.50	0.41	0.52	0.47	0.46	0.26	0.58	0.53	0.61	0.52	0.64	0.52
RV50	0.69	0.35	0.65	0.29	0.68	0.32	0.63	0.25	0.69	0.35	0.72	0.36	0.68	0.31

Probcons, Probalign, Clustal Omega, and T-Coffee programs were more successful than others in the precision of alignments measured as a percentage of the corresponding reference datasets. In fact, it underscoring the effectiveness of these methods to in producing accurate and reliable alignments, thereby making them a valuable training for further optimization through the GAMSA framework.

V. Conclusion

This paper addresses the challenges faced in accurately aligning protein sequences with complex amino acid structures. The proposed Genetic Algorithm for Multiple Sequence Alignment (GAMSA) algorithm offers a novel solution to these challenges. The GAMSA approach aims to enhance the alignment accuracy and efficiency for sequences with intricate structures. To validate the effectiveness of GAMSA, nine MSA tools were tested on protein database. The alignments were evaluated using metrics calculated with MSASuite. The results demonstrate that the proposed approach performs well on sequences with a large number of residues, especially those with complex amino acid structures. GAMSA significantly improves efficiency and reduces computational costs compared to traditional methods. As discussed in the introduction, GAMSA offers a less complex approach that can be applied to both short and long sequences.

References

[1] L. Santus, E. Garriga, S. Deorowicz, A. Gudyś, and C. Notredame, “Towards the accurate alignment of over a million protein sequences: Current state of the art,” *Curr. Opin. Struct. Biol.*, vol. 80, p. 102577, Jun. 2023, doi: 10.1016/J.SBI.2023.102577.

[2] B. Dou *et al.*, “Machine Learning Methods for Small Data Challenges in Molecular Science,” *Chem. Rev.*, vol. 123, no. 13, pp. 8736–8780, Jul. 2023, doi: 10.1021/ACS.CHEMREV.3C00189/ASSET/IMAGES/MEDIUM/CR3C00189_0026.GIF.

[3] C. Gaad, M.-A. Chadi, M. Sraitih, and A. Aamouche, “Exploring Reinforcement Learning Methods for Multiple Sequence Alignment: A Brief Review”, doi: 10.1051/bioconf/20237501004.

[4] Q. Bani Baker, R. A. Al-Hussien, and M. Al-Ayyoub, “Accelerating Multiple Sequence Alignments Using Parallel Computing,” *Computation*, vol. 12, no. 2, p. 32, 2024, doi: 10.3390/computation12020032.

[5] V. Ranwez *et al.*, “Strengths and Limits of Multiple Sequence Alignment and Filtering Methods To cite this version : HAL Id : hal-02535389 Chapter 2 . 2 Strengths and Limits of Multiple Sequence Alignment and Filtering Methods,” pp. 0–36, 2020.

[6] T. Paruchuri, G. R. Kancharla, and S. Dara, “Solving multiple sequence alignment problems by using a swarm intelligent optimization based approach,” vol. 13, no. 1, p. 11591, 2023, doi: 10.11591/ijece.v13i1.pp1097-1104.

[7] M. K. Ibrahim, U. K. Yusof, T. Abdalla, E. Eisa, and M. Nasser, “Enhanced Genetic Method for Optimizing Multiple Sequence Alignment,” 2023.

[8] G. Benedetti, “RNA Secondary Structure Prediction Using a Genetic Algorithm with a Selection Method Based on Free Energy Value and Topological Index,” pp. 1–16, 2024.

[9] E. V. Korotkov and D. O. Kostenko, “Application of the MAHDS Method for Multiple Alignment of Highly Diverged Amino Acid Sequences,” *Int. J. Mol. Sci. 2022, Vol. 23, Page 3764*, vol. 23, no. 7, p. 3764, Mar. 2022, doi: 10.3390/IJMS23073764.

[10] M. T. Pervez *et al.*, “IvisTMSA: Interactive visual tools for multiple sequence alignments,” *Evol. Bioinforma.*, vol. 11, no. March, pp. 35–42, 2015, doi: 10.4137/EBO.S18980.

[11] Y. Ye and B. G. Iovino, “Protein Embedding based Alignment,” *Authorea Prepr.*, May 2023, doi: 10.22541/AU.168534397.72964200/V1.

[12] T. Wehning, “Improving Performance of Multiple Sequence Alignment through Maximal Exact Match Identification,” 2023.

[13] A. Alqahtani and M. Almutairy, “Evaluating the Performance of Multiple Sequence Alignment Programs with Application to Genotyping SARS-CoV-2 in the Saudi Population,” *Comput. 2023, Vol. 11, Page 212*, vol. 11, no. 11, p. 212, Nov. 2023, doi: 10.3390/COMPUTATION11110212.

[14] C. D. McWhite and M. Singh, “Vector-clustering Multiple Sequence Alignment: Aligning into the twilight zone of protein sequence similarity with protein language models,” *bioRxiv*, p. 2022.10.21.513099, Apr. 2023, doi: 10.1101/2022.10.21.513099.

[15] M. Bioinspired, C. Yang, M. K. Ibrahim, U. Kalsom Yusof, T. Abdalla Elfadil Eisa, and M. Nasser, “Bioinspired Algorithms

for Multiple Sequence Alignment: A Systematic Review and Roadmap," *Appl. Sci. 2024, Vol. 14, Page 2433*, vol. 14, no. 6, p. 2433, Mar. 2024, doi: 10.3390/APP14062433.

[16] X. Wan, K. Liu, W. Qiu, and Z. Kang, "An Assembly Sequence Planning Method Based on Multiple Optimal Solutions Genetic Algorithm," *Mathematics*, vol. 12, no. 4, 2024, doi: 10.3390/math12040574.

[17] Z. Y. Wang and C. Lu, "An integrated job shop scheduling and assembly sequence planning approach for discrete manufacturing," *J. Manuf. Syst.*, vol. 61, pp. 27–44, Oct. 2021, doi: 10.1016/J.JMSY.2021.08.003.

[18] S. K. Yadav, S. Kumar Jha, S. Singh, P. Dixit, S. Prakash, and A. Singh, "Optimizing Multiple Sequence Alignment using Multi-Objective Genetic Algorithms," *2022 Int. Conf. Decis. Aid Sci. Appl. DASA 2022*, pp. 113–117, 2022, doi: 10.1109/DASA54658.2022.9765131.

[19] F. A. Tuğcan KORAK, "Multiple sequence alignment quality comparison in T-Coffee, MUSCLE and M-Coffee based on different benchmarks," *Cumhur. Sci. J.*, vol. 42, no. 1, pp. 30–37, 2021.

Mubashir Imam (MS Software Engineering, GCU Faisalabad) contributed to designing the genetic algorithm framework in GAMSA-Align, focusing on optimization for protein sequence alignment. His research interests include deep learning, cybersecurity, data science, and genetic algorithms.

Mueed Ahmed Mirza (MS Computer Science, Riphah International University) provided expertise in bioinformatics and data science, aiding in methodology analysis and algorithm validation. His research interests span bioinformatics, machine learning, data science, and networking.

Wasif Ali (MS Computer Science, COMSATS Islamabad) worked on developing and optimizing the algorithmic structure for efficient sequence alignment. His interests are in deep learning, computer vision, and the analysis of algorithms.

Hafiz Haseeb Tasleem (MS Data Science, Riphah International University) with a focus on big data, natural language processing (NLP), and deep learning and supported data processing and analysis to ensure scalability and effectiveness in GAMSA-Aligns results.

MUHAMMAD USMAN KARIM *(MS Computer Science. Riphah International University) with a focus on AI, deep learning, healthcare, Component Based Software Engineering, and Requirement Engineering.*