Solving multiple sequence alignment problems by using a swarm intelligent optimization based approach

Tirumala Paruchuri[1,2], Gangadhara Rao Kancharla[2], Suresh Dara[3]
[1]Department of Management Information System, Jazan University, Kingdom of Saudi Arabia
[2]Department of Computer Science and Engineering, Acharya Nagarjuna University, Andhra Pradesh, India
[3]School of Technology, Woxsen University, Hyderabad, Telangana, India

Article Info

Article history:

Received Apr 26, 2022
Revised Aug 9, 2022
Accepted Sep 5, 2022

Keywords:

Evolutionary computation technique
Multiple sequence alignment
Optimization
Pairwise alignment
Particle swarm optimization

ABSTRACT

In this article, the alignment of multiple sequences is examined through swarm intelligence based an improved particle swarm optimization (PSO). A random heuristic technique for solving discrete optimization problems and realistic estimation was recently discovered in PSO. The PSO approach is a nature-inspired technique based on intelligence and swarm movement. Thus, each solution is encoded as “chromosomes” in the genetic algorithm (GA). Based on the optimization of the objective function, the fitness function is designed to maximize the suitable components of the sequence and reduce the unsuitable components of the sequence. The availability of a public benchmark data set such as the Bali base is seen as an assessment of the proposed system performance, with the potential for PSO to reveal problems in adapting to better performance. This proposed system is compared with few existing approaches such as deoxyribonucleic acid (DNA) or ribonucleic acid (RNA) alignment (DIALIGN), PILEUP8, hidden Markov model training (HMMT), rubber band technique-genetic algorithm (RBT-GA) and ML-PIMA. In many cases, the experimental results are well implemented in the proposed system compared to other existing approaches.

Corresponding Author:

Suresh Dara
School of Technology, Woxsen University
Hyderabad, Telangana, India
Email: darasuresh@live.in

1. INTRODUCTION

The methanesulfonic acid (MSA) is one of the most challenging and powerful tasks in solving computational problems. Evaluating the biological sequences will helps in MSA process. MSA is defined by the combination of amino acids and nucleotide sequences. Primary and secondary structures will be useful in understanding the structure of a given sequence. MSA helps in finding the phylogenetic distance for a particular sequence. In computational biology, the main task is to predict the order of the structure of a given molecule [1], [2] and also MSA makes it more useful to find the structure. In computational biology, the MSA will be useful for solving a variety of tasks. So, this is the main reason why researchers are more focused on solving MSA problems effectively. The primary purpose of MSA is to differentiate sequences with similar inactive properties. However, if possible, not only distinguish rows with inactive characters, but also compare them with other characters. In bioinformatics, the main problem in MSA is knowing the relationship between phylogenetic and genetics. So far, various methods have been used to overcome this problem.

Recently, the nature inspired computing is very popular to solving kind of optimization problems, particularly in MSA problems [3]. Especially, many algorithms are successfully applied to solving multiple sequence alignment problem. Lalwani *et al.* [4] introduced a new approach as two-level strategy for solving MSA problem. A swarm intelligence-based optimization approach is used to solve MSA problem in [5].

To effectively solve MSA problems, a dynamic programming (DP) approach is useful. In DP, the large domain scoring feature is used to update results for MSA issues. In 1970 the DP solution to the MSA problem was proposed by Needleman and Wunsch [6]. Assuming that the number and length of sequences increase, the DP approach will break easily. However, considering the time complexity and number of sof the dynamic program approach also increases with increasing sequence length and number. So, the MSA problem will led to NP-hard. Therefore, the major aim is to tackle MSA issues in an systematic manner i.e., similarities need to be developed among the sequences with less complexity. Another way to deal MSA problems is to use a progressive alignment approach. However, it will be useful in solving because it is less complex [6], [7]. According to this method, we first assume a similar sequence and then refer to a dynamic set of those sequences. CLUSTAL W [8] is a classic approach in progressive methods.

Another approach is an iterative procedure used to solve MSA problems. It starts with an initial alignment, filtering the frequent solutions until no possible improvement. Here, the decision does not depend on the initial alignment. The target of the iteration process is to enhance the quality of alignment. Simulated annealing is one type of iterative or stochastic method. A simulated annealing with particle swarm optimization (PSO) is proposed for MSA problem [9], which mainly focus on global search by PSO and to avoid the trapping in local optima is incorporated with simulated annealing.

Evolutionary algorithms (EAs) usually rely on population-based approach mainly focus on global search. In MSA, EA will be useful in randomization, and the next EA step will be applied to the sequence to improve similarity. Sequence alignment by genetic algorithm (SAGA) [10], rubber band technique-genetic algorithm (RBT-GA) [11], and several other genetic algorithm (GA) algorithms are based on MSA. Using reference sets 2 and 3, Taheri and Zomaya [11] solved 34 issues from the Bali database. In EA, local optimal solutions achieve low variance, and this is considered a major disadvantage. DP is used in the proposed local optimum solution method. We have an operator called inbuilt elitism in PSO. The proposed method of elitism and initial operator modification will help to solve the MSA problem effectively.

Another swarm intelligent based approach is proposed by inspired from the artificial fish algorithm. Dabba *et al.* [12] proposed multi objective artificial fish swarm algorithm called MOAFS for solving MSA problem. Here, they proposed two fitness functions to maintain diversity in exploration and exploitation in the search space. Chaabane *et al.* [13] proposed a hybrid approach for solving MSA as PSO with Tabu search called PSO-TS. The PSO used effectively explore the search space and Tabu search used for produce the global best solutions in high quality. A bird swarm align algorithm [14] also proposed for solving MSA problem. Brainstorm optimization based algorithm with a dynamic population size is proposed for solving MSA problem in [15], [16].

A hybrid approach is proposed in [17] which is combination of PSO and artificial bee colony (ABC) optimization technique called hybrid algorithm of artificial bee colony and particle swarm optimization (HABC-PSO). It integrates the Tent chaos search with opposition-based learning where they incorporate with recombination operator technique, to gain best solutions by using this recombination operator. A multi objective ABC proposed [18] for MSA problem, used two objective functions as one is sum of pairs and another one is entropy. Another ABC based three level multi objective approach for ribonucleic acid (RNA) multiple structural alignment problem [19]. A new kind approach based on flower pollination algorithm (FPA) [20] is proposed for solving MSA problem, this metaheuristic approach used for finding the alignment score based maximizing the sum of pairs and column score.

This article organizing as, required method PSO is discussed in section 2. The proposed method of PSO was discussed in step by step briefly along with illustration of decoding and encoding in section 3. In section 4, we discussed the proposed method experimental studies as well as comparison with other methods in briefly. At last, we conclude the paper in section 5.

2. METHODS

Many evolutionary computation techniques have been successes fully used for solving optimization problem. In this article we have adopted the one of the swarm intelligent based techniques is PSO. Finding the right solution in a large search space is very useful. In 1995, Kennedy and Eberhart [21] was inspired by schools of fish or birds. A binary version of PSO was introduced in [22] for MSA problem. We used same kind of approach in out experimental study. The (1) and (2) describe velocity and position updates.

$$V_{\mathrm{i,d}}(\mathrm{t}+1) = \mathrm{w} * V_{\mathrm{i,d}}(\mathrm{t}) + \mathrm{c1} * \mathrm{r1} * \left(pbest - X_{\mathrm{i,d}}\right) + \mathrm{c1} * \mathrm{r2} * \left(pbest_{\mathrm{gbest,d}} - X_{\mathrm{i,d}}\right) \tag{1}$$

$$X_{\mathrm{i,d}}(\mathrm{t}+1) = X_{\mathrm{i,d}}(\mathrm{t}) + V_{\mathrm{i,d}}(\mathrm{t}+1) \qquad (2)$$

Where *X* is the current position, *r1* and *r2* are represented as random vectors in [0,1], *V* is shown as particle velocity, *c1* and *c2* represented as accelerating constants, t is considered as iteration, and w represents the inertial weight.

3. PROPOSED APPROACH

This section going to discuss the proposed method of PSO and explained the process in step by step. The following steps are explained as: initializing the position of the particle, how representation as particle and how updating the particle position through velocity and (1), (2). Given a complete description of proposed fitness function, and calculation of alignment score. The explanation of this particular steps is discussed below.

3.1. Initial generation

The development of some initial solutions is considered the main goal. However, initial solutions can be developed by providing gaps between the residues. The following example shows the generation of the initial population. Figure 1(a) shows the initial MSA. From each pair we find pairwise alignment using the Wunsch and Needleman algorithms. In Figure 1(b) shows the initial pair (1, 2). Then, pairwise alignment for the initial pair (1, 2) need to be find, which was shown in Figure 1(c). Suppose if the problem is with 4 sequences, then *4*(4-2)/2=6* pairs are considered. So pairwise alignment must be found in the same way for each pair. Now random permutation from 1 to *N* must be generated. For example, if we have 4 sequences, any permutation from 1 to 4 can be generated. If every permutation of (3, 4, 1 and 2) is generated then Figure 1(d) shows the alignment process. *K* solutions can be generated by *K* times of random permutations from *1* →*N*. At this stage, the initial population is developed.

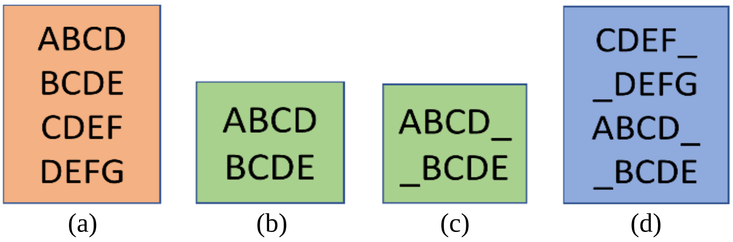


Figure 1. An illustration of generate an initial solution (a) initial MSA, (b) initial pair (1, 2), (c) pairwise alignment for the initial pair (1, 2), and (d) alignment process

3.2. Solution representation as a particle

Each and every solution was represented in PSO as particle.

$$X_{\mathrm{i},} = \; X_1, X_2, \ldots\ldots X_{\mathrm{d}} \qquad \text{where } 1 \leq \mathrm{I} \geq \mathrm{N}.$$

Where *d* is the dimension size in a particle, *N* represents the number of particles. Out of the 200 solutions we record one solution and explain how we can encode the MSA problem into a particle. Figure 2 shows the initial solution.

Binary encoding schematic-in this part of the schematic, place 1 at the gap position and 0 at the protein sequence position. This is shown in Figure 2 showing how the initial solution is encoding a particle. In each column we see decimal values from bottom to top with binary encoded values. So, *X1=(1, 0, 0, 8, 4,* and *2*) is a solution for this particle representation. Therefore, in MSA, the number of columns is equal to the number of dimensions in the particle. In MSA, with the encoding scheme, new particles are only developed after the initial solution generation.

3.3. Fitness function

MSA utilize the weighted sum of pair method (WSPM) function to calculate the fitness measurement. In this, score for every column is equal to the computation of product of every pair symbol score.

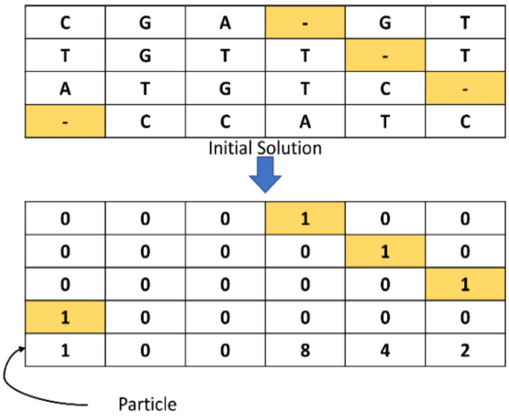
$$S = \sum_{i=1}^{L} S_{l,} \tag{3}$$

$$S_{l,} = \sum_{i=1}^{N-1} \sum_{j=1+1}^{N} \left(W_{ij}\, Cost\, (A_{i,}A_{j,})\right) \tag{4}$$

The overall alignment score is computed against the column scores through (3). Where l indicates the number of columns in the total alignment, *S* indicates the MSA cost. *N* represents the sequence count, S_l represents the value for the lth column of length l, W_{ij} represents the weights of the rows for *i* and *j*. Its main purpose is to explain the variance between the two sequences.

W_{ij}=(Miss)/(TL)

Where, *Miss* count of mismatch characters in the alignment, *TL* lengh of total align. *Cost*(A_i, A_j) is a function use to calculate the alignment score *og* two aligned sequences A_i and A_j. Here, there are four possible cases. (1) When $A_i \neq$ "_" and $A_j \neq$ "_" then cost (A_i, A_j) was explained from the acceptable mutation's matrix percentage. The (2) suppose A_i= "_" and A_j= "_" then cost (A_i, A_j)=*0*. The (3) and (4) at last, cost (A_i, A_j) is the cost function, which considers the summing of substitution costs like deletions or insertions takes place if A_i= "_" *and* $A_j \neq$ "_" or *viseversa* then cost (A_i, A_j)=*1*.



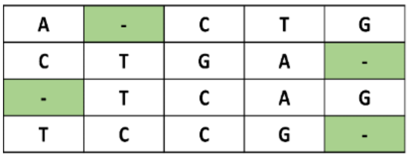
C	G	A	-	G	T
T	G	T	T	-	T
A	T	G	T	C	-
-	C	C	A	T	C

0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1
1	0	0	0	0	0
1	0	0	8	4	2

Figure 2. The binary encoding process from initial solution

3.4. An alignment calculation

If 5 solutions were taken in the initialization process and in each dimension, every particle is assigned with initial velocity i.e., 0. Later in the initialization process, out of a total of 100 solutions, 1 solution has been shown in Figure 3. Based on encoding scheme, such solution turned into a particle (2, 8, 0, 0, 5). For every particle, we find the particle in the similar manner.



A	-	C	T	G
C	T	G	A	-
-	T	C	A	G
T	C	C	G	-

Figure 3. Initial alignment

Suppose the remaining particles solutions are *P1=(2, 8, 0, 0, 5)*, *P2=(0, 4, 4, 0, 0)*, *P3=(3, 5 1, 2, 4) P4=(4, 2, 1, 8, 1)*, and *P5=(9, 1, 5, 3, 4)*. The calculate the pbest values as *pbest1=(8, 4, 2, 1,1)*, *pbest2=(6, 2, 4, 3, 8)*, *pbest3=(5, 3, 7, 0, 2)*, *pbest4=(2, 2,8, 0, 4)*, and *pbest5=(6, 1, 8, 2, 2)*. Later, find the fitness value of each particle by using (3) suppose after calculating the fitness value for five particles the values like *Fit (P1)=13*, similarly 19, 18, 7, 15,13, and 19 for respective particle fitness values. Then, the *Fit (pbest1)=13*, *Fit (pbest2)=19*, *Fit (pbest3)=18*, *Fit (pbest4)=7*, and *Fit (pbest5)=15*.

Update the particle position and velocity: Through (1) and (2), we need to update the position and velocity.

$$V_{\mathrm{i,d}}(\mathrm{t}+1) = 0.691 * 0 + 2.15 * 0.35 * (2-2) + 2.22 * 0.46 * (4-2)$$
$$=0+0+2.042$$
$$=2.042$$

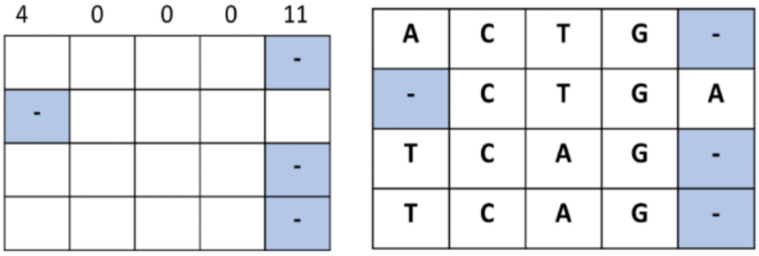
Now calculate of the new position

$$X_{\mathrm{i,d}}(\mathrm{t}+1) = X_{\mathrm{i,d}}(\mathrm{t}) + V_{\mathrm{i,d}}(\mathrm{t}+1)$$
$$=2+2.042= 4.042$$
$$=4$$

Here, updated position of floor value is considered due to the presence of discrete valued function in encoding scheme. However, 4 is the updated position, then every particle in each dimension gets updated in the similar manner. Hence, we will get new particles like *P1=(4, 0, 3, 0, 2), P2=(4, 2, 4, 1, 2), P3=(2, 4, 0, 1, 2), P4=(3, 6, 0, 1, 4),* and *P5=(1, 5, 0, 1, 4)*. Also, fitness value for new particles can be *foundFit (P1)=44, Fit (P2)=5, Fit (P3)=12, Fit (P4)=30,* and *Fit (P5)=10*. Later, *pbest* value can be updated through the algorithm which is provided in Algorithm 1. The gap can be provided based on decoding scheme, which is explained in Figure 4(a). Later, protein sequences can be allotted in specific positions to produce the final MSA. Figure 4 (b) determines the final MSA.

Algorithm 1. The PSO approach for MSA problem

```
1  Initialize the random solutions as particles with X and V values for N solutions
2  Calculate fitness value for each particle using (3), find pbest
3  Begin
4       for i=1 to N;
5       If (Fit (Pi)>Fit (pbesti)//update pbest
6               pbesti=Pi;
7       End If
8       End for
9  repeats from 3 until finite number of iterations
10 End
```



4	0	0	0	11
				-
-				
				-
				-

(a)

A	C	T	G	-
-	C	T	G	A
T	C	A	G	-
T	C	A	G	-

(b)

Figure 4. A sample solution in a iteration (a) a decoding scheme and (b) final updated MSA

4. RESULTS AND DISCUSSION

In this section, we have implemented the propose approach and the results are examined and then compared with the few existing methods. The parameter setting of PSO are considered as N=200, $V_{max}=2^{Number\ of\ sequences}$, $V_{min}=0$, $w=0.758$, $c_1=2.15$, $c_2=2.22$, r_1 and r_2 are a random value in [0,1]. Here, the performance of our proposed approach is compared with few existing approaches using different reference datasets. The compared results presented in Tables 1 and 2.

4.1. Comparative performance of proposed method

In our experiments, we tool a benchmark dataset from publicly available database Benchmark alignment dataBASE (BaliBASE) [23], [24]. Randomly 16 datasets have picked from the database. So, 12 datasets are from reference set 2 and remaining 4 datasets are from reference set 3. The existing results are taken from the literature RBT-GA [11]. Also the experimental results compared with literature [25].

4.2. Experimental results in BAliBASE reference 2 datasets

From such reference, 12 datasets were extracted. All of these sets are different with respect to the terms of number and length. Our proposed method is well compared with various existing approaches like ML-PIMA, DIALIGN, rubber band technique-genetic algorithm (RBT-GA), hidden Markov model training (HMMT) and PILEUP-8 for evaluating the proposed method performance. Table 1, presented the results reference 2 datasets, and it determines that the proposed method performed well in 12 datasets and existing methods in 7 test cases. Also, the proposed approach does not achieve best solution within 5 test cases but it just closely related towards the best solution. Finally, Table 2 explains that the entire performance of the proposed system performed well than the other existing methods in reference 3 datasets. The proposed approach score was 0.708 whereas other approaches are an average less than 0.3. only one approach getting good results than our proposed approach that is RBT-GA which reported as 0.83.

Table 1. Result comparison in reference 2 dataset

Datasets	DIALI	PILEUP-8	HMMT	RBT-GA	ML-PIMA	Proposed Approach
2trx	0.734	0.87	0.739	0.982	0.702	0.787
1r69	0.675	0.45	0	0.9	0.675	0.251
1sbp	0.043	0.177	0.214	0.778	0.054	0.817
1csy	0	0.114	0	0.735	0	0.797
1wit	0.724	0.476	0.641	0.825	0.463	0.757
1tvxA	0	0.345	0.276	0.891	0.241	0.534
1aboA	0.384	0	0.724	0.812	0.22	0.838
1idy	0	0	0.353	0.997	0	0.559
1havA	0	0.493	0.194	0.792	0.238	0.817
1uky	0.216	0.562	0.395	0.625	0.306	0.719
1tgxA	0.63	0.318	0.622	0.835	0.543	0.886
1ubi	0	0	0.053	0.795	0.129	0.799
Avg score	**0.283**	**0.317**	**0.35**	**0.83**	**0.297**	**0.708**

4.3. Experimental results in BAliBASE reference 3 datasets

Within reference set 3, we have considered 4 datasets *lidy, luky, lr69* and *lubi* out of 12 datasets. Table 2 describes the experimental results on reference 3 test cases that performed better and 1 test case did not perform well but it almost closes to the best. However, Table 2 shown the proposed approach was outperformed with better average solutions than existing methods. Table 2 shows that the comparative results on [3] dataset with few existing algorithms. The results are considered as average score, the proposed approach claim as 0.514, at same time other approaches reported RBT-GA as 0.395, and the next highest was ML-PIMA as 0.263, all other approaches were reported very less score comparing with our proposed approach.

Table 2. Result comparison on reference 3 dataset

Datasets	DIALI	PILEUP-8	HMMT	RBT-GA	ML-PIMA	Proposed approach
1idy	0	0	0.227	0.548	0	0.623
1uky	0.139	0.083	0.037	0.35	0.148	0.411
1r69	0.524	0	0	0.374	0.905	0.425
1ubi	0	0.268	0.366	0.31	0	0.598
Avg score	**0.165**	**0.087**	**0.157**	**0.395**	**0.263**	**0.514**

5. CONCLUSION

This article is mainly discussed about an improved PSO for solving the alignment problem. Also, experiments have conducted in multiple times to evaluate the performance of proposed system. Meanwhile in PSO approach, an initial operator is used for evaluating. Also, initial operator is improved with more efficacy and dominance. The weighted pair function was used in the proposed system for fitness calculation. The powerful datasets have taken from public database i.e., BaliBASE with 2.0 version, which was explained the efficacy of the RBT-GA algorithm. Later, better solutions of Bali score have taken for comparison with various existing methods. In most of the test results, the proposed system performed best and, in few cases, the performance is slightly lacked but it almost closes to the solution. The major reason in utilizing the improved initial operator was to improve the proposed system efficacy. Therefore, we claim that our proposed system shown better results than the existing approaches.

REFERENCES

[1] A. R. Amorim, G. F. D. Zafalon, A. de G. Contessoto, C. R. Valêncio, and L. M. Sato, "Metaheuristics For multiple sequence alignment: A systematic review," *Computational Biology and Chemistry*, vol. 94, Oct. 2021, doi: 10.1016/j.compbiolchem.2021.107563.

[2] Y. Zhang, Q. Zhang, J. Zhou, and Q. Zou, "A survey on the algorithm and development of multiple sequence alignment," *Briefings in Bioinformatics*, vol. 23, no. 3, May 2022, doi: 10.1093/bib/bbac069.

[3] A. F. Ali and A.-E. Hassanien, "A survey of metaheuristics methods for bioinformatics applications," *Applications of intelligent optimization in biology and medicine*, pp. 23–46, 2016, doi: 10.1007/978-3-319-21212-8_2.

[4] S. Lalwani, H. Sharma, A. Verma, and R. Kumar, "Efficient discrete firefly algorithm for Ctrie based caching of multiple sequence alignment on optimally scheduled parallel machines," *CAAI Transactions on Intelligence Technology*, vol. 4, no. 2, pp. 92–100, Jun. 2019, doi: 10.1049/trit.2018.1040.

[5] Á. Rubio-Largo, L. Vanneschi, M. Castelli, and M. A. Vega-Rodríguez, "Swarm intelligence for optimizing the parameters of multiple sequence aligners," *Swarm and Evolutionary Computation*, vol. 42, pp. 16–28, 2018, doi: 10.1016/j.swevo.2018.04.003.

[6] 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, no. 3, pp. 443–453, Mar. 1970, doi: 10.1016/0022-2836(70)90057-4.

[7] J. Armstrong *et al.*, "Progressive Cactus is a multiple-genome aligner for the thousand-genome era," *Nature*, vol. 587, no. 7833, pp. 246–251, Nov. 2020, doi: 10.1038/s41586-020-2871-y.

[8] J. D. Thompson, D. G. Higgins, and T. J. Gibson, "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice," *Nucleic Acids Research*, vol. 22, no. 22, pp. 4673–4680, 1994, doi: 10.1093/nar/22.22.4673.

[9] L. Chaabane, "A hybrid solver for protein multiple sequence alignment problem," *Journal of Bioinformatics and Computational Biology*, vol. 16, no. 4, Aug. 2018, doi: 10.1142/S0219720018500154.

[10] C. Notredame, "SAGA: sequence alignment by genetic algorithm," *Nucleic Acids Research*, vol. 24, no. 8, pp. 1515–1524, Apr. 1996, doi: 10.1093/nar/24.8.1515.

[11] J. Taheri and A. Y. Zomaya, "RBT-GA: a novel Meta heuristic for solving the multiple sequence alignment problem," *Bmc Genomics*, vol. 10, no. 1, pp. 1–11, 2009.

[12] A. Dabba, A. Tari, and D. Zouache, "Multiobjective artificial fish swarm algorithm for multiple sequence alignment," *INFOR: Information Systems and Operational Research*, vol. 58, no. 1, pp. 38–59, Jan. 2020, doi: 10.1080/03155986.2019.1629782.

[13] L. Chaabane, A. Khelassi, A. Terziev, N. Andreopoulos, M. A. de Jesus, and V. V. Estrela, "Particle swarm optimization with tabu search algorithm (PSO-TS) applied to multiple sequence alignment problem," in *Advances in Multidisciplinary Medical Technologies-Engineering, Modeling and Findings*, Cham: Springer International Publishing, 2021, pp. 103–114.

[14] H. A. Rehman, K. Zafar, A. Khan, and A. Imtiaz, "Multiple sequence alignment using enhanced bird swarm align algorithm," *Journal of Intelligent and Fuzzy Systems*, vol. 41, no. 1, pp. 1097–1114, Aug. 2021, doi: 10.3233/JIFS-210055.

[15] J. J. Pujari and K. K. Pavan, "Multiple sequence alignment based on enhanced brainstorm optimization algorithm with dynamic population size (EBSODP)," *Annals of the Romanian Society for Cell Biology*, pp. 10033–10042, 2021.

[16] J. J. Pujari and K. K. Pavan, "The brain storm dynamic technique DBSO-MSA for efficiently resolving multiple sequence alignment," *Journal of Mobile Multimedia*, Mar. 2022, doi: 10.13052/jmm1550-4646.18411.

[17] N. Altwaijry, M. Almasoud, A. Almalki, and I. Al-Turaiki, "Multiple Sequence alignment using a multiobjective artificial bee colony algorithm," in *2020 3rd International Conference on Computer Applications and Information Security (ICCAIS)*, Mar. 2020, pp. 1–6, doi: 10.1109/ICCAIS48893.2020.9096734.

[18] M. Kaya, A. Sarhan, and R. Alhajj, "Multiple sequence alignment with affine gap by using multi-objective genetic algorithm," *Computer Methods and Programs in Biomedicine*, vol. 114, no. 1, pp. 38–49, Apr. 2014, doi: 10.1016/j.cmpb.2014.01.013.

[19] S. Lalwani, H. Sharma, and K. Deep, "An implementation of three-level multi-objective ABC algorithm for RNA multiple structural alignment," *Recent Advances in Computer Science and Communications*, vol. 13, no. 1, pp. 68–76, Mar. 2020, doi: 10.2174/2213275912666181128121104.

[20] A. M. Hussein, R. Abdullah, and N. AbdulRashid, "Flower pollination algorithm with profile technique for multiple sequence alignment," in *2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology (JEEIT)*, Apr. 2019, pp. 571–576, doi: 10.1109/JEEIT.2019.8717513.

[21] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, pp. 1942–1948, doi: 10.1109/ICNN.1995.488968.

[22] Y. Du, J. He, and C. Du, "A novel binary particle swarm optimization for multiple sequence alignment," in *Intelligent Computing Theories and Application*, Springer International Publishing, 2019, pp. 13–25.

[23] A. Bahr, "BAliBASE (Benchmark Alignment dataBASE): enhancements for repeats, transmembrane sequences and circular permutations," *Nucleic Acids Research*, vol. 29, no. 1, pp. 323–326, Jan. 2001, doi: 10.1093/nar/29.1.323.

[24] J. D. Thompson, P. Koehl, R. Ripp, and O. Poch, "BAliBASE 3.0: Latest developments of the multiple sequence alignment benchmark," *Proteins: Structure, Function, and Bioinformatics*, vol. 61, no. 1, pp. 127–136, Jul. 2005, doi: 10.1002/prot.20527.

[25] M. I. Menhas, M. Fei, L. Wang, and X. Fu, "A novel hybrid binary PSO algorithm," in *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, 2011, pp. 93–100.

BIOGRAPHIES OF AUTHORS

Tirumala Paruchuri she obtained her Masters in Computer Applications (Sri Krishna Devaraya University, Anantapur, India) and Masters in Technology-Computer Science and Engineering (Jawaharlal Nehru Technological University, Hyderabad, India). She started her career in teaching with St. Mary's College of Engineering and Technology, Hyderabad, India and presently teaching in the department of Management Information System, Jazan University, Jazan, Kingdom of Saudi Arabia and also pursuing her research (Ph.D.) in Acharya Nagarjuna university, Guntur, India. Her research interests are data mining, machine learning, cloud computing, evolutionary computing, and big data analytics. She can be contacted at email: tirumalaparuchuri@gmail.com.

Gangadhara Rao Kancharla Professor Gangadhara Rao did his B.Tech in Electrical and Electronics Engineering later he did his M.Tech in Computer Science. He was awarded his Ph.D. in Computer Science for the thesis titled “Analytical and Game theoretical approach of TCP SYN flooding for web server”. Several research scholars working under him in the areas of AI&ML. Cloud computing, big data and Manets. He has published many research articles in reputed journals. He has advisory member of many reputed international conferences. He can be contacted at email: kancherla123@gmail.com.

Suresh Dara he is currently working as a Professor in B V Raju Institute of Technology, Narsapur, Medak-502313, India. He has completed his Ph.D. from Indian Institute of Technology (ISM), Dhanbad, India in 2015, completed his Masters in CST from Andhra University, India in 2008. He has published many research articles in reputed journals, international conferences, Books and Book chapters. He served as reviewer for many reputed journals. His current research interests are soft computing, machine learning, deep learning, evolutionary computation, and bioinformatics. He has committee member of many reputed international conferences and workshops. He can be contacted at email: darasuresh@live.in.