

Chapter 8

Particle Swarm Optimization with Tabu Search Algorithm (PSO-TS) Applied to Multiple Sequence Alignment Problem



Lamiche Chaabane , Abdeldjalil Khelassi , Andrey Terziev , Nikolaos Andreopoulos , M. A. de Jesus , and Vania Vieira Estrela 

8.1 Introduction

Sequence alignment is paramount to molecular sequence analysis. It can help to build a phylogenetic tree of related DNA sequences or to predict the function/structure of unfamiliar protein sequences by aligning them with others whose function/structure is already acknowledged. Sequence alignment establishes an alignment of two or more sequences to maximize the similarities amid them [1]. This NP-hard problem [2, 3] can benefit from computational intelligence (CI) algorithms in reasonable processing time.

In recent years, metaheuristic procedures aided in producing approximate solutions for the MSA dilemma. Their primary rationale is to commence by an initial solution and ameliorate the MSA through a series of iterations until the solution does not become better any longer. This category of methods embraces tabu search (TS) [4], genetic algorithm (GA) [5], simulated annealing (SA) algorithm [6], ant colony (AC) algorithm [7], particle swarm optimization (PSO) [8], artificial bee colony (ABC) algorithm [9], and so on.

This research study brings in a hybrid line of attack named PSO-TS algorithm to devise an approximate solution to the MSA impasse combining the best characteristics of the PSO and the TS schemes.

L. Chaabane (Deceased)

Computer Science Department, University of M'sila, M'sila, Algeria

A. Khelassi (✉)

Abou Baker Belkaied University Of Tlemcen, Tlemcen, Algeria

A. Terziev · N. Andreopoulos · M. A. de Jesus · V. V. Estrela

Department of Telecommunications, Federal Fluminense University (UFF), RJ, Brazil

e-mail: vania.estrela.phd@ieee.org

© Springer Nature Switzerland AG 2021

A. Khelassi, V. V. Estrela (eds.), *Advances in Multidisciplinary Medical Technologies — Engineering, Modeling and Findings*, https://doi.org/10.1007/978-3-030-57552-6_8

103

The remainder of this document is organized as follows: Section 8.2 presents a brief literature review of the previous works pertinent to the PSO-TS framework. Section 8.3 depicts the basic concepts of the PSO and the TS metaheuristics. The details about the components of the PSO-TS algorithm are in Sect. 8.4. Simulation outcomes appear in Sect. 8.5, together with some discussions. Section 8.6 suggest improvements to the PSO-TS algorithm. Finally, this study ends with the comments from Sect. 8.7.

8.2 Literature Review

Among the significant multiple sequence approaches, some of them employ iterative alignment aligners. Rias et al. [4] applied a tabu search variant relying on the COFFEE objective function to find better results for the multiple sequences. In [5], the authors developed an amended GA with an intelligent selection strategy to resolve the multiple biological sequence dilemma. Proper alignments are obtained compared with those done by classical GA.

In Ref. [8], the authors applied PSO and Clustal X aligner to produce better results than those from Clustal X alone. An ABC algorithm for solving the MSA issue appears in [9]. In Ref. [10], V. Cutello et al. presented an immune-inspired algorithm (IMSA) to tackle the MSA dilemma utilizing ad hoc mutation operators. Simulation results on some BaliBASE v.1.0 instances show that IMSA is superior to other aligners such as PRRP, CLUSTAL X, SAGA, DIALIGN, PIMA, MULTIALIGN, and PILEUP8. An efficient method by using a multi-objective GA (MSAGMOGA) to discover optimal alignments is proposed in [11]. Experiments on the BaliBASE 2.0 database confirmed that MSAGMOGA obtained better results than MUSCLE, SAGA, and MSA-GA methods.

Recently, in [12], the authors announced a novel multi-objective evolutionary scheme called hybrid multi-objective ABC (HMOABC), which hinges on swarm intelligence for the MSA dilemma. Experimental results on some instances coming from the BaliBASE 3.0 database showed that the HMOABC is an auspicious approach for solving the MSA impasse. Reference [13] suggests a better-quality Chemical Reaction Optimization (CRO) algorithm for the MSA problem. In their work, the authors inserted an intelligent diversification mechanism in the initialization process to help the CRO algorithm to travel around the search space globally. Simulation results confirmed the superiority of this investigative effort over others.

8.3 Preliminaries

8.3.1 Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is a metaheuristic (aka nature-inspired) optimization algorithm conceived by Kennedy and Eberhart [14] that had as its inspiration the bird flocking behavior. Rather than the hard optimization formulation, there is a new recast problem. This different formulation consists of a population of answers entitled particles, where the pair of features position and velocity characterize each particle. At each iteration, the particle's locus is updated in accordance with its personal best position and the best solution of the swarm. The subsequent equations govern the evolution of the population:

$$V^{(k+1)} = w.V^{(k)} + c_1.rand_1.(pbest^{(k)} - X^{(k)}) + c_2.rand_2.(gbest^{(k)} - X^{(k)}), \text{ and} \quad (8.1)$$

$$X^{(k+1)} = X^{(k)} + V^{(k+1)}, \quad (8.2)$$

where for each particle:

X is its position

V is its velocity

w represents the inertia weight

$pbest$ is its best place

$gbest$ is the global best position of the swarm

$rand1, rand2$ are random values within $[0, 1]$

$c1, c2$ are positive constants regulating the impact of the personal optimum as well as the global best solutions on the search course, respectively

k is the iteration number

The process stops after reaching a predefined number of iterations $Itmax$.

8.3.2 Tabu Search Optimization (TS)

TS appeared the first time in Glover's manuscript (1986) [15] to unravel a wide range of hard optimization problems. TS starts with a random tentative answer and appraises the fitness function for the given solution. Then all possible neighbors of the given solution are generated and evaluated. To overcome the local optimum limit, TS saves the better local neighbors (after searching, examining, and, if necessary, discarding them) in a Tabu List (TL). Moreover, TS can exploit an aspiration rule to make the whole search processing continue or to devise a diversified guided search tactic to other promising regions.

8.4 Proposed Method

PSO's main drawback is its premature convergence and local optimum problem. To evade this limit, TS is incorporated as a local search procedure to enrich the global best solution and to apply the diversification mechanism to guide the search to other promising regions of the search space. Then, the novel hybrid technique, called PSO-TS, consists of strong cooperation among PSO and TS. It makes full use of the exploration ability of PSO and the exploitation ability of TS.

In our MSA problem, PSO-TS works as follows:

1. PSO modifies both particle positions and velocities after establishing the initial population with the recommended swarm initialization strategy.
2. After that, a new global best solution ($gbest^*$) emerges from the improvement of the current global best solution ($gbest$) via the TS procedure.
3. This iterative process ends when a stopping condition is reached.

The PSO-TS hybridization strategy has a flowchart of is along these lines (Fig. 8.1):

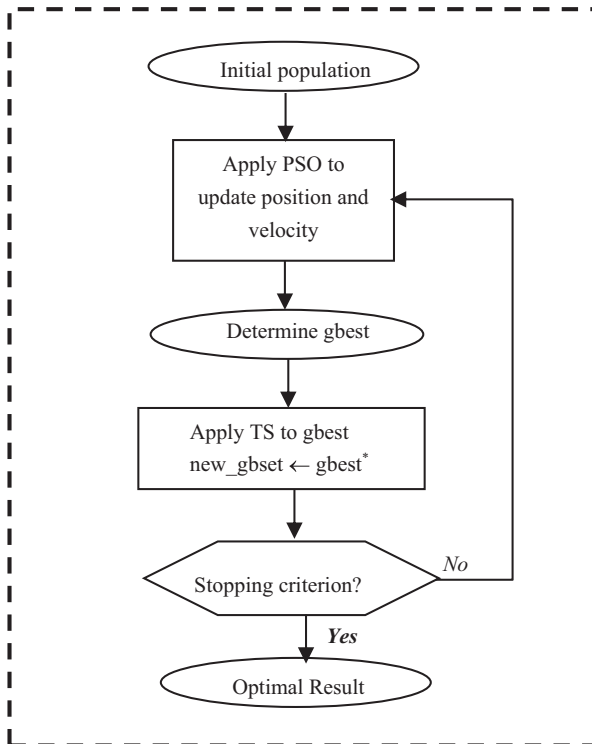


Fig. 8.1 PSO-TS algorithm flowchart

8.4.1 Components

Solution Encoding Each solution (particle) entails a set of vectors, whose elements indicate the position of the randomly interleaved gaps in the different sequences waiting for alignment. Here, the sequences should have the same length L (the typical L value is 1.2 times of the most prolonged sequences) [16]. Figure 8.2 brings in an example of this encoding scheme.

Swarm Initialization The multiple small-popsizes initialization strategy (MSPIS) [17] renders the initial population and ensures its diversity. Every time, there is a small number of chromosomes by a small-popsizes tactic. The best two individuals are selected into the initial population until the number of chromosomes equal to the present population size. A summary of the detailed MSPIS steps becomes.

- Step1.* Generate a small number of chromosomes through the small-popsizes initial method.
- Step2.* Calculate the mean value of the small number population fitness value, $Meanpop_i$.
- Step3.* All the chromosomes whose fitness values exceed $Meanpop_i$ go into the initial population.
- Step4.* Repeat steps 1–3 until the number of chromosomes equal to the present population size.

This initialization method delivers an approximate optimal solution as near as possible.

Objective Function Habitually, the objective function appraises the alignment quality in mathematical terms. For that, the score apportioned to each particle (alignment) consists of the summing of the scores (SP) of the alignment of each pair of sequences [18].

The score of each pair of sequences comprises the sum of the score assigned to the match of each pair of symbols. A substitution matrix, such as the PAM205 matrix, can achieve this effect. The objective function for an alignment A with k sequences is

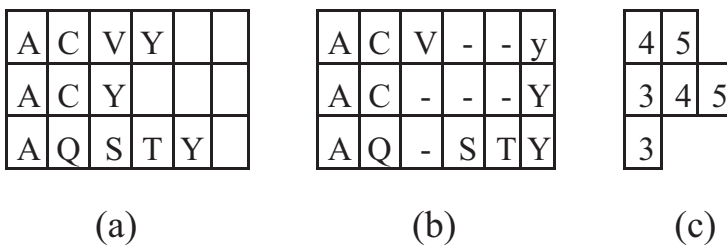


Fig. 8.2 Sequence alignment: (a) Set of sequences to align, (b) insertion of gaps, and (c) solution representation

$$\text{Score}(A) = \sum_{i=1}^{k-1} \sum_{j=i+1}^k S(A_i, A_j),$$

where $S(A_i, A_j)$ stands for the alignment score among two aligned sequences, A_i and A_j .

Particle Move Operator This operator is necessary to update the position of the particle in the search space. Each swarm element movement is contingent on its current location and the same as the leader. In this work, the distance between the particle and the leader amounts to the proportion of matching gaps in the sequences as

$$\text{Distance} = \frac{\text{matching gaps}}{\text{total gaps}}.$$

This research introduced the crossover operator from [16] to transport each particle toward the leader. It entails dividing the current alignment into two segments according to a randomly picked crossover point from the range $[1, \text{Distance} * L]$. After that, the best-produced child replaces the current particle.

8.4.2 TS Components of the MSA Problem

Cost Function It evaluates the quality of each candidate solution. In this research study, the authors retain the same one from the PSO algorithm.

Initial Solution In the suggested PSO-TS methodology, the TS works as a local search process to ameliorate the global best solution (*gbest*) yielded by the PSO algorithm. Then, the PSO output becomes an initial solution for the TS procedure.

Generation of Neighbors In order to generate a set of neighbors of the current solution, the authors apply a simple but effective strategy, which works as follows:

- (i) Firstly, it picks a random amino acid from a randomly chosen sequence in the alignment and checks whether one of its neighbors is a gap.
- (ii) If this is the circumstance, the algorithm swaps the elected amino acid with a gap neighbor.
- (iii) If both neighbors are gaps, then one of them is picked randomly [18].

Figure 8.3 portrays this mechanism.

Tabu List Update This list stores previously visited solutions to help the algorithm evade being trapped in local optima. Here, the user, depending on the size of the problem instances, predefines the tabu list length (TLL). Besides, the authors opted to modify the TLL value dynamically by increasing or diminishing it with a

A	C	N	-	K	T	V	Y
A	C	-	K	-	-	Y	-
A	Q	-	-	S	-	V	Y

A	C	N	-	K	T	V	Y
A	C	-	-	K	-	Y	-
A	Q	-	-	S	-	V	Y

Fig. 8.3 Neighbors generation strategy

prefixed constant X . Such a mechanism helps to attain some other solution and to overcome the cycling problem.

Stopping Condition The authors chose the most straightforward and most commonly used stopping criterion, i.e., to terminate upon reaching a predefined number of iterations.

The pseudo-code of our TS is given below:

Pseudo-code of TS algorithm

Step 0: Generate an initial alignment A_{init} .
 Initialize $ITMAX, N_s, TLL$
 Set $TL \leftarrow \emptyset, A_{curr} = A_{init}, A_{best} = A_{init}$

Step 2: Iterate the following steps for $ITMAX$ iterations

Step 2.1: Generate N_s neighboring alignments
 $A_i (i=1, 2, \dots, N_s)$ of A_{curr}

Step 2.2: $A_{best} \leftarrow Best_neighbor(A_1, A_2, \dots, A_{N_s})$
 If $(A_{best} \notin TL)$ and $S(A_{best}) > S(A_{curr})$ then
 $A_{curr} \leftarrow A_{best}$
 Else try next test alignment and go to step 2.1
 $TL \leftarrow TL \cup A_{curr}$
 If TL is full, remove the oldest solution from TL

Step 2.3: If $S(A_{curr}) > S(A_{best})$ then
 $A_{best} \leftarrow A_{curr}$

Step 3: Return A_{BEST} and $S(A_{best})$

After the description of both the PSO and TS algorithms, the global framework of the PSO-TS procedure is as follows:

8.5 Simulation and Results

The PSO-TS algorithm is implemented using JCreator version 3.5 and a personal computer with a 2.66 GHz Intel Pentium IV processor. A set of tests verified and validated the PSO-TS scheme, where the objective is to compare the PSO-TS method with other published works, including the IMSA approach [10] and the

PSO-TS Pseudo-code

1. Apply MSPIS strategy to generate an initial swarm
2. Fix NG value *// Number of generations*
3. Set $j \leftarrow 1$
4. While ($j \leq NG$) do
 - 4.1. Determine the leader particle value $gbest$
 - 4.2. For each particle in the population do
 - a) Evaluate the distance between $gbest$ and the particle
 - b) Select a crossover point randomly
 - c) Apply particle move mechanism
 - d) Assign to the current particle the best-produced child
 - End for
 - 4.3 Update $gbest$ *// Calculate the new leader particle gbest*
 - 4.4 $New_gbest \leftarrow TS(gbest)$ *// Apply TS to improve gbest*
 - 4.5 Insert New_gbest into the current population
 - 4.6. $j \leftarrow j + 1$
5. End while
6. Return $gbest$ value and its correspondent particle. *// Output the result.*

BPSO algorithm [19]. The BaliBASE SP score (SPS) results for both small and large datasets taken from the BaliBASE database [20] are portrayed in Tables 8.1 and 8.2.

The subsequent procedure provides the SP score [21]: for a candidate alignment (individual) of N sequences encompassing M columns, then the i -th column within the alignment becomes $A_{i1}, A_{i2}, \dots, A_{iN}$. A value of p_{ijk} characterizes each pair of residues A_{ij} and A_{ik} . If $p_{ijk} = 1$, it means that the candidate alignment residues A_{ij} and A_{ik} have been aligned with each other as far as the reference alignment is concerned. Otherwise, $p_{ijk} = 0$. Consequently, the score for the i -th column turns out to be

$$S_i = \sum_{j=1, j \neq k}^N \sum_{k=1}^N p_{ijk}.$$

For the candidate alignment, the overall SPS becomes

$$SPS = \sum_{i=1}^M \frac{S_i}{Mr},$$

with the number of columns in the reference alignment given by Mr while S_{ri} is the score S_i for the i -th column for the reference alignment. The values of SPS lie in the interval $[0.0, 1.0]$. High SPS values point toward closer likeness with the BaliBASE reference alignment [21].

Table 8.1 PSO-TS vs. IMSA: SPS comparative results for the BaliBASE test sets

Instance	N	IMSA [10]	PSO-TS
laboA	5	0.759	0.910
45lc	5	0.773	0.853
9rnt	5	0.954	0.948
kinase	5	0.644	0.716
2cba	5	0.754	0.754
lppn	5	0.987	0.989
2myr	4	0.285	0.433
left	4	0.880	0.928
ltaq	5	0.946	0.949
lubi	17	0.897	0.907
kinase	18	0.905	0.915
lidy	27	0.854	0.849
Average		0.810	0.846

Table 8.2 PSO-TS algorithm versus BPSO approach: SPS comparative results for the BaliBASE test sets

Instance	N	BPSO [19]	PSO-TS
lidy	5	0.7394	0.8492
45lc	5	0.7973	0.8532
lkrn	5	0.9984	0.9988
kinase	5	0.7064	0.7165
lpii	5	0.7987	0.7980
5ptp	5	0.9328	0.9410
lajsA	5	0.3528	0.3750
glg	5	0.8324	0.9241
ltaq	5	0.7633	0.9491
Average		0.7690	0.8227

From Tables 8.1 and 8.2, it is clear that comparing with IMSA and BPSO, the average SPS values of PSO-TS algorithm are more significant than or comparable with those obtained by the techniques stated above for the short, medium, and long sequences. In addition, significant improvements are observed in test cases with a large number of sequences, which shows that our approach can work well with large problem sizes.

8.6 Future Trends

Other metaheuristics and multi-objective optimization schemes can aid in unraveling this bioinformatics impasse [22–24].

As perspective innovations of this work, the authors contemplate (i) integration of other mechanisms within the neighborhood generation step or (ii) intensification/

diversification strategies in TS core to improve the quality of multiple alignments. In addition, the authors will use other powerful PSO variants as a part of this hybridization tactic to find a better solution for the MSA problem. A comparison of the proposed method with some other state-of-the-art techniques such as Clustal W, SAGA, or MULTALIGN, with other benchmarks, is possible to verify its enormous potential [25–27].

Deep learning (DL) has been in the spotlight as a potent tactic that makes noteworthy signs of progress when it comes to solving the issues plaguing the artificial intelligence. Nevertheless, numerous caveats, e.g., local minima trapping, inferior performance, and elevated computational time, still arise while applying DL to MSA. So, global optimization procedures like differential search algorithms can aid DL deployments to reach the best outcome and data [28–31].

8.7 Conclusion

The MSA problem is one of the primary techniques utilized in computational biology, e.g., genomic annotation, homology searches, gene regulation networks, protein structure prediction, and functional genomics, to name a few. Among MSA bottlenecks, there is the alignment of more than a pair of biological sequences, which is an NP-hard optimization problem.

This research showcases the PSO-TS approach, which is a novel hybrid algorithm constructed on metaheuristics developed to tackle the multiple sequence alignment (MSA) problem. The PSO-TS framework combines the PSO benefits, particularly its simplicity in implementation and its inexpensive computational processing time, with the best tabu search algorithm features, i.e., its local upgrading practice. These improvements enrich the global best solution by augmenting the diversification process. The results summarized in this paper show the superior capability of our hybrid approach compared to some other literature works.

References

1. J.D. Thompson, J.E. Thierry, O. Poch, Rapid scanning and correction of multiple sequence alignments. *Bioinformatics* **19**, 1155–1161 (2003)
2. T. Jiang, L. Wang, On the complexity of multiple sequence alignment. *J. Comput. BioI.* **1**, 337–378 (1994)
3. C.H. Papadimitriou, K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity* (Dover Publications, New York, 1998)
4. T. Riaz, Y. Wang, K.B. Li, Multiple sequence alignment using tabu search, in *Proceedings of 2nd Asia-Pacific Bioinformatics Conference (APBC)*, Dunedin, New Zealand, (2004), pp. 223–232
5. J.T. Hornig, L.C. Wu, C.M. Lin, B.H. Yang, A genetic algorithm for multiple sequence alignment, in *Proceedings of LNCS*, (2005), pp. 407–420

6. M. Hernández-Guía, R. Mulet, S. Rodríguez-Pérez, A new simulated annealing algorithm for the multiple sequence alignment problem. The approach of polymers in a random media. *Phys. Rev. E* **72**, 1–7 (2005)
7. L. Chen, L. Zou, J. Chen, An efficient ant colony algorithm for multiple sequences alignment, in *Proceedings of the 3rd International Conference on Natural Computation (ICNC '07)*, (2007), pp. 208–212
8. F. Xu, Y. Chen, A method for multiple sequence alignment based on particle swarm optimization, in *ICIC 2009, LNAI 5755*, (2009), pp. 965–973
9. X. Lei, J. Sun, X. Xu, L. Guo, Artificial bee colony algorithm for solving multiple sequence alignment, in *Proceedings of 2010 IEEE Fifth International Conference on BIC-TA*, (2010), pp. 337–342
10. V. Cutello, G. Nicosia, M. Pavone, I. Prizzi, Protein multiple sequence alignment by hybrid bio-inspired algorithms. *Nucleic Acids Res.* **39**(6), 1980–1992 (2011)
11. M. Kayaa, A. Sarhanb, R. Alhajj, Multiple sequence alignment with affine gap by using multi-objective genetic algorithm. *Comput. Methods Prog. Biomed.* **114**, 38–49 (2014)
12. R.L. Alvaro, A.V.R. Miguel, L.G.A. David, Hybrid multi-objective artificial bee colony for multiple sequence alignment. *Appl. Soft Comput.* **41**, 157–168 (2016)
13. R.K. Yadav, H. Banka, An improved chemical reaction-based approach for multiple sequence alignment. *Curr. Sci.* **112**(3), 527–538 (2017)
14. J. Kennedy, R. Eberhart, Particle swarm optimization. *Proc. IEEE Int. Conf. Neural Netw.* **4**, 1942–1948 (1995) Perth
15. F. Glover, Tabu search: Part 1. *ORSA J. Comput.* **1**(3), 190–206 (1989)
16. P.F. Rodriguez, L.F. Nino, O.M. Alonso, Multiple sequence alignment using swarm intelligence. *Int. J. Comput. Intell. Res.* **3**(2), 123–130 (2007)
17. C.Z. Ahn, R.S. Ramakrishna, A genetic algorithm for shortest path routing problem and the size of populations. *IEEE Trans. Evol. Comput.* **6**(6), 566–579 (2002)
18. C. Lamiche, A hybrid solver for protein multiple sequence alignment problem. *J. Bioinforma. Comput. Biol.* **16**(4), 1–20 (2018)
19. H.X. Long, W.B. Xu, J. Sun, W.J. Ji, Multiple sequence alignment based on a binary particle swarm optimization algorithm, in *Proceedings of Fifth International Conference on Natural Computation*, (2009), pp. 265–269
20. J.D. Thompson, F. Plewniak, O. Poch, A comprehensive comparison of multiple sequence alignment programs. *Nucleic Acids Res.* **27**(13), 2682–2690 (1999)
21. S.S. Sathya, S. Kuppaswami, K.S. Babu, Nomadic genetic algorithm for multiple sequence alignment (MSANGA). *Int. J. Adapt. Innov. Syst.* **1**, 44 (2009)
22. N. Razmjooy, V.V. Estrela, *Applications of Image Processing and Soft Computing Systems in Agriculture*. IGI Global. (2019). <https://doi.org/10.4018/978-1-5225-8027-0>
23. N. Razmjooy, V.V. Estrela, H.J. Loschi, A study on metaheuristic-based neural networks for image segmentation purposes, in *Data Science Theory, Analysis and Applications*, Taylor and Francis, Abingdon, UK, (2019)
24. M.A. de Jesus, V.V. Estrela, O. Saotome, D. Stutz, Super-resolution via particle swarm optimization variants, in *Biologically Rationalized Computing Techniques for Image Processing Applications. Lecture Notes in Computational Vision and Biomechanics*, ed. by J. Hemanth, V. Balas, vol. 25, (Springer, Cham, 2018). https://doi.org/10.1007/978-3-319-61316-1_14
25. C. Zambrano-Vega, A.J. Nebro, J. García-Nieto, J.F. Montes, Comparing multi-objective metaheuristics for solving a three-objective formulation of multiple sequence alignment. *Prog. Artif. Intell.* **6**, 195–210 (2017)
26. S. Chatterjee, P. Barua, M. Hasibuzzaman, A. Iftiea, T. Mukharjee, S.S. Nova, A hybrid genetic algorithm with chemical reaction optimization for multiple sequence alignment, in *2019 22nd International Conference on Computer and Information Technology (ICCIT)*, (2019), pp. 1–6
27. F. Naznin, R.A. Sarker, D. Essam, Progressive alignment method using genetic algorithm for multiple sequence alignment. *IEEE Trans. Evol. Comput.* **16**, 615–631 (2012)

28. Deshpande A, Patavardhan P, Estrela VV, Razmjooy N. (2020) Deep learning as an alternative to super-resolution imaging in UAV systems. In: Estrela V.V., Hemanth J., Saotome O., Nikolakopoulos G., Sabatini R. (eds), *Imaging and Sensing for Unmanned Aircraft Systems*, Vol. 2, 9, 177–212, Stevenage: The Institution of Engineering and Technology
29. R. Jafari, M.M. Javidi, M.K. Rafsanjani, Using deep reinforcement learning approach for solving the multiple sequence alignment problem. *SN Appl. Sci.* **1**, 1–12 (2019)
30. C. Zhang, W. Zheng, S.M. Mortuza, Y. Li, Y.A. Zhang, DeepMSA: Constructing deep multiple sequence alignment to improve contact prediction and fold-recognition for distant-homology proteins. *Bioinformatics* (2019)
31. S.N. Yousoff, A. Baharin, A. Abdullah, A review on optimization algorithm for deep learning method in bioinformatics field, in *2016 IEEE EMBS Conference on Biomedical Engineering and Sciences (IECBES)*, (2016), pp. 707–711