A Novel Bio-Geography Based Approach for Multiple Sequence Alignment

Rohit Kumar Yadav
Assistant Professor, Department of Computer Science, IITM Janakpuri, New Delhi, India.
rohit.ism.123@gmail.com

Abstract

In this paper, an improved Biogeography Based Optimization (BBO) is applied for aligning multiple sequences. Biogeography Based Optimization is a nature inspired technique which is based on species migration one to other habitat due to climate. Here we used improve migration operator in place of conventional migration operator. However, there are some deficiencies in solving complicated problems, due to low population diversity and slow convergence speed in the later stage. To overcome these drawbacks, we propose an improved BBO algorithm integrating a new improved migration operator. The improved migration operator simultaneously adopts more information from other habitats, maintains population diversity and preserves exploitation ability. The performance of the proposed method has been tested on publicly available benchmark datasets (i.e. Bali base) with some of the existing methods such as VDGA, MOMSA and GAPAM. It has been observed that, the proposed method perform better and/or competitive in most of the cases.

Keywords: Multiple Sequence Alignment (MSA), Bio-Geography Based Optimization (BBO), Migration Operator, Diversity.

1 Introduction

More than 3 amino acid sequence or protein sequence alignment at a time is called MSA. MSA is most important tool to solve biological problems. We can solve lots of problem in biology by the use of MSA. MSA help to predict the secondary and tertiary structure of RNA and proteins [11, 8]. We can reconstruct phylogenetic trees by the use of MSA which can predict the function of an unknown amino acid by aligning its sequences with some other known functions. We can also find similarity of the sequences by the use of MSA, which can helps to define similarity in functions and structures [2, 4]. In order for a MSA to be valid entire sequences in the multiple alignments must have a common origin. The goal of MSA is to maximize the matching of protein or amino acid as far as possible [6]. Therefore MSA is an important problem in bioinformatics to study of genetic and phylogenetic relationship. There are several method to solve MSA problem in past.

Multiple sequence alignment can be solved and achieve optimal alignment by the use of Dynamic programming (DP). DP uses a scoring function which contains large domain. In 1970, Needleman and Wunsch in the article [17] proposed the use of dynamic programming algorithm to solve the problem of two sequence alignment. But problem behind the use of Dynamic Programming (DP) is when the number and length of sequence are increase its complexity also increases in exponential manner. Then the MSA problem becomes to NP-hard. Since complexity is main constraints to solve any problem by the computer. So we have to maximize the matching of protein or amino acid sequence in limited time or less complexity. This is the major reason researchers switch to another methods.

MSA problem can be also solved by progressive method. The Progressive approach takes less complexity in terms of time and space for solving MSA problem [24, 9]. According to progressive alignment method initially align more similar sequences after that it incrementally align more divergent sequence or group of sequence in the initial alignment. The standard representative of progressive methods is CLUSTALW [25]. In first step, according to this approach we have to assign weight of each pair of sequences in a partial alignment. We assign small-weight of most similar sequences and big-weight for most divergent sequences. After that we take substitution matrix which defines the score between two residues of protein sequence based on similarity. Two types of gap have been introduced in third step. First one is residue-specific gap and second one is locally residue gap penalties. In fourth step, where gap have been introduced in early position receive locally reduced gap penalties to encourage the opening gap at these positions.

These four steps are integrated into CLUSTALW which is freely available. Progressive alignment method performs better for MSA package in terms of accuracy and time. Even that this method has some limitation. The problem behind this method is dependency on initial alignment and choice of scoring scheme. In other words we bound that to align more similar sequence in initial stage. If we have not aligned more similar sequences in initial stage then the solution may be trapped in local optima. An Iterative method is another option for solving multiple sequence alignment.

An iterative method does not depend on initial alignment because it starts with initial alignment and improve the solutions per iteration until no more improvement possible. The main objective of the iterative approach for MSA is to globally improve the quality of a sequence alignment. There are some iterative and stochastic approaches for MSA (as example, simulated annealing [14, 16]). HMMT [7] is based on a simulated annealing process. The problem behind these methods solution may be trapped in local optima.

Evolutionary algorithms are population based algorithm. According to these algorithms, we generate random initial population in the first step. Next step, we apply some operators to modify initial population for next generation. We repeatedly use these operators until reach the global optimum. When using EAs for MSA, an initial generation is generated by random manner, and then the steps of an EA are applied to improve the similarities among the sequences. There are some evolutionary computations for MSA [3, 5, 12, 13, 18]. There are some other genetic algorithm (GA) based methods for MSA, such as SAGA [18], GA-ACO [15], MSA-EC [20], MSA-GA [10], RBT-GA [21], GAPAM [27], VDGA[28] and MOMSA[29]. We define methodology of some algorithm MSA problem based on Genetic algorithm (GA). In SAGA, the initial generation is generated randomly. According to SAGA, 22 different operators are used to gradually improve the fitness of MSA. But the problem behind SAGA is time complexity due to repeated use of fitness function. RBT-GA is also a GA based method, combined with the rubber band technique (RBT), to find optimal protein sequence alignments [23]. RBT [22] is an iterative algorithm for sequence alignment using a DP table. The authors [29] solved 56 problems from reference sets 1,2,3,4 and 5 of the benchmark Bali base 2.0 dataset and Bali base 3.0 dataset. The drawbacks of these evolutionary methods are also local optima due to poor diversity of the solutions.

1.1 Motivation and Contributions

In the domain of biology, Multiple Sequence Alignment (MSA) is the most crucial to solve numerous standards problems such as structure prediction, Phylogenetic property etc. According to the open literature, the MSA is still open challenging problem. Hence, we motivate to solve MSA problem using improved version of BBO. However, this paper achieves the following contributions.

a. We first proposed a method to improve migration operator in BBO and then used it in MSA for maintaining diversity of the solutions.

b. The results obtained in experimental analysis are better in terms of time factor. In addition, we provide a comparison table which claims that our method is better than existing competitive solutions in terms of matching score.

1.2 Structure of the paper

After presenting satisfactory introduction in Section 1, we discuss Bio-geography based optimization (BBO) as preliminaries of our work in Section 2. The Section 3 addresses our proposed method and the description of test data set appears in Section 4. In Section 5, we present experimental analysis of our proposed work. We drawn conclusion of this paper in Section 6 and complete it with several related references.

2 Bio-Geography Based Optimization

BBO [19] was designed by emigration and immigration of species in one to other habitats. In the BBO algorithm, candidate solutions are called habitats (or islands). Each feature in a solution represented by a habitat is called a suitability index variable (SIV), while the goodness of a habitat is measured by the habitat suitability index (HSI). Habitats with a high HSI can support more species, whereas low HSI habitats support only a few species. Poor habitats can improve their HSI by accepting new features from more attractive habitats in the evolution process. In BBO, there are two main operators: migration, mutation. The migration operator is a probabilistic operator that can randomly modify SIVs based on the immigration rate λi and emigration rate μi. Both λi and μi are functions of the number of species in the i-th habitat (Hi). In the original BBO algorithm, for mathematical convenience, μi and λi are assumed to be linear with the same maximum values, which means that the immigration rate λi and emigration rate μi are linear functions of the number of species. The linear migration model for the i-th habitat (Hi) can be calculated as

$$\lambda i = I*(1-ni) /n$$

$$\mu i = E*ni/n \quad (1)$$

Where E is the maximum possible emigration rate, I is the maximum possible immigration rate, ni is the number of species in i-th Habitat and n is the maximum Number of species. In BBO, the migration operator is a probabilistic operator used to randomly adjust each habitat Hi by sharing features among them. The probability that Hi is modified is proportional to its immigration rate λi , while the probability that the source of the modification comes from Hj is proportional to the emigration rate μj . The migration equation is expressed as

$$Hi\ (SIV) = Hj\ (SIV) \quad (2)$$

Where Hi (SIV) denotes the feature (SIV) of the i-th habitat Hi. As Simon stated, the migration operator merely migrates SIVs from one solution to another, and does not involve reproduction of "children" [19].

Cataclysmic events can cause a species count to differ from its equilibrium value, thereby suddenly changing a Habitat's HSI. We model this sudden operation in BBO as mutation. The SIVs of the i-th habitat Hi can be randomly modified by the mutation operator according to the habitat's priori probability pi . The mutation probability mi of the i-th habitat Hi is expressed as

$$mi = mmax * (1- Pi /Pmax) \quad (3)$$

Where m max is a user-defined parameter and P max = max (pi), i = 1, 2. . . N. In the BBO mutation operator, an SIV in each habitat is randomly replaced by a new feature, randomly and probabilistically generated in the entire solution space, which tends to increase population diversity.

3 Proposed Method

3.1 Habitat Representation

In BBO each solution is represented as Habitat

C	G	A	_	G	T
A	T	G	T	C	_
T	G	T	T	_	T
_	C	C	A	T	C

Fig. 1. Initial Solution

$$X_i = (X_i^1, \ldots, X_i^d, \ldots, X_i^n) \qquad \forall 1 \le i \ge N \quad (4)$$

Where N is the number of Habitat. In Initialization State First put the gap in our given MSA is randomly. The initial solution has been given in fig. 1.

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

Fig. 2. Encoding Scheme

Binary encoding Scheme: - In the encoding scheme put 1 in position of Gap and put 0 in the position of protein sequences. Fig. 2 shows an encoding of initial solution. After that we are taking decimal value of this binary encoded value from bottom to top of each column. Hence our Habitat representation of this solution is X1= (1, 0, 0, 8, 2, 4). Hence the number of columns in the MSA is the number of features in Habitat. Now according to this manner, we can generate 100 number of solution putting Gap in MSA. Hence we can find 100 Habitat in initialization.

3.2 Fitness function

The Sum of Pair is used to measure fitness of multiple sequence alignments. Here, each column in an alignment is scored by summing the product of the scores of each pair of symbols. The score of the entire alignment is then summed over all column scores by using (5) and (6)

$$W = \sum_{i=1}^{P} W_i \text{ where} \qquad W_i = \sum_{i=1}^{N-1} \sum_{k=i+1}^{N} Cost(A_i, A_k) \quad (5)$$

Here, W is the cost of multiple sequence alignments. P is the length (columns) of the alignment; Wi is the cost of the ith column of P length. N is the number of sequences. Cost (Al , Ak) is the alignment score between two aligned sequences Al and Ak . When Al ≠ "_" and Ak ≠"_" Then cost (Al , Ak) is determined from the percentage of acceptable mutations matrix (PAM). Also when Al = "_" and

Ak =“_” then Cost (Al, Ak) = 0. Finally, the cost function Cost (Al, Ak) includes the sum of the substitution costs of the insertion/deletions when Al = “_” and Ak ≠ “_” or Al ≠ “_” and Ak = “_” using a model with affine gap penalties as shown in Eqn. 6

$$Z = Q + Ar. \quad (6)$$

Here, Z is the gap penalty, Q is the cost of opening a gap, r is the cost of extending the gap , and A is the number of the gap. In this paper gap penalties (gap opening penalty is −5 and the gap extension penalty is −0.40.

3.3 Illustration

Suppose initial problem is given in Fig. 3.

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

Fig. 3. Initial MSA

Smax =100 (Maximum Number of species in a Habitat)
I=1 (Maximum Immigration rate)
E=1(Maximum Emigration rate)
And mmax = 0.2 (Maximum mutation probability)
Now, we are taking 5 solutions after putting random gap. Hence according to our encoding scheme our Habitats are given below:
H1 = (4, 8, 0, 2, 1) H2 = (2, 4, 0, 1, 2) H3 = (4, 10, 1, 0, 2) H4 = (8, 2, 4, 0, 2) H5 = (4, 1, 0, 0, 4)
We can find Habitat Suitability index (HSI) of each Habitat using fitness function equation 5 & 6.
F1 = 11 F2 = 20 F3 = 13 F4 = 10 F5 = 5
We can find best and worst Habitat suitability index (since our objective is maximize the score).
Hence best = 20 and worst = 5
According to HSI of each Habitat, we can find number of species in each Habitat.

Number of species = Smax * (Hk – worst)/(best – worst)
n1 = Smax * (H1 – worst)/(best –worst)
=100* (11-5) / (20-5)
=100* 0.4
=40
Similarly we can find n2, n3, n4, n5
n2= 100 n3 = 66 n4= =33 n5 = 0
After that we can calculate λ (Immigration rate) and μ (Emigration rate) of each Habitat.
λk =I * (1- nk /Smax)
=1*(1- 40/60)
=60/100
=0.6
Similarly λ2 = 0 λ3 =.34 λ4 = .67 λ5 = 1
Also we can calculate μk (Emigration rate)

μk = 1 - λk

μ1 = 0.4 μ2 = 1 μ3 = 0.66 μ4 = 0.33 μ5 = 0

Now, according to migration process

Hi(SIV) = Hj(SIV) + F*(Hp1(SIV) – Hp2(SIV))
Since Immigration rate of λ5 =1 and Emigration rate of μ2 =1. Hence feature of H2 migrate in features of H5.

Now, we can select two random number p1 and p2. Suppose p1 = 4 and p2 = 3.

F is scaling factor between 0 and 1.

Let F = 0.5, now we can generate a random number between 1 and 5.
Suppose we have generated a random number is 2. Hence according to improved migration operator we can update the habitat features.
H5(2)= H2(2) + F*(H4(2) – H3(2))
=4 +0.5 *(2-10)
=4-4
=0
Hence Habitat H5 = (4,0,0,0,4)
Now, in mutation process
Suppose selected habitat is H5. We can generate randomly index in Habitat H5. Suppose it is 5.After that according to mutation process we can generate one random number between 1 and range of features in Habitat.
Suppose it is 11 and replace to index 5 of Habitat 5. Our Habitat H5 becomes (4, 0, 0, 0, 11).
Hence after complete process of BBO for 1st iteration, Our Habitats are:
H1 = (4, 8, 0, 2, 1) H2 = (2, 4, 0, 1, 2) H3 = (4, 2, 1, 0, 2) H4 = (8, 2, 4, 0, 2) H5 = (4, 0, 0, 0, 11)
We can put the gap in each column according to our decoding scheme. This is given in Fig. 4.

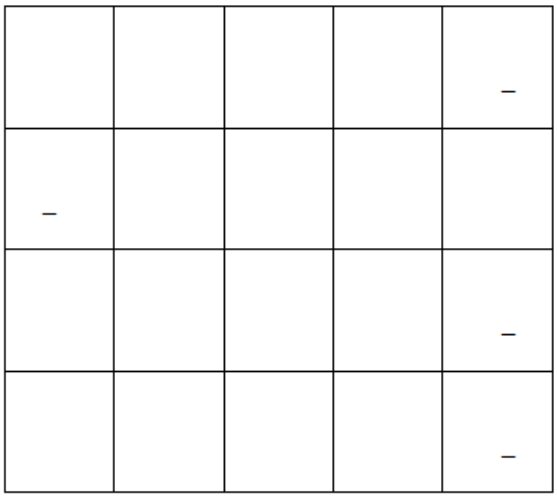

Fig. 4. Decoding Scheme

After complete this process put protein sequence in vacant place. Hence, final solution of initial MSA is given in Fig. 5.

A	C	T	G	_
_	C	T	G	A
T	C	A	G	_
T	C	C	G	_

Fig. 5. Final solution

We have seen that alignment of MSA is very efficient in only one generation due to use of improved migration operator.

4 Test Dataset

We have tested a large number of datasets from Bali base benchmark database to check the quality of our approach. Bali base version 1.0 [26] contains 142 reference alignment which keeps more than 1000 sequences. Bali base version 2.0 [1] is an extended version of Bali base version 1.0. Bali base version 2.0 contains 167 reference alignments which keeps more than 2100 sequences. Bali base version 2.0 contains eight reference sets. Each reference keeps different type of sequences. Small number of equidistance sequence contains in reference set 1. Totally different or unrelated sequence contains in reference set 2. Reference set 3 contains a pair of divergent sub-families. Long terminal extension sequence contains in reference set 4. Reference 5 contains large internal insertions and deletions. Lastly Reference 6-8 contain test case problems where the sequence are repeated and the domains are inverted. Bali score is a score measure the quality of algorithm. Bali score compare between manually alignment sequence (which is available on Bali base version 2.0) and alignment which is come from some existence method. Range of Bali score is 0 to 1. If the manually alignment file and our output file is same then score is 1. If the manually alignment file and our output file is totally different then score is 0. It gives the value between 0 and 1 according to similarity between Bali base manually alignment file and our output file.

5 Experimental Analyses

In this section, firstly, we compare IBBOMSA with the recently proposed multiple sequence alignment algorithms based on evolutionary algorithms, including VDGA[28], GAPAM[27] and MOMSA[29] to prove its dominance. After that, we also compare the performance of IBBOMSA with many well-liked aligners. In this paper, IBBOMSA is coded in C language and implemented on the personal computer in linux platform. Spelling, Language and Capitals

5.1 Effect of Improved operator in BBO

The BBO algorithm was invented for immigration and emigration of species between habitats in multidimensional search space. Each habitat represents a solution. In traditional BBO migration features of good solution appears in poor solution as a new feature while still remaining in good solution. Since this feature may exist in several number of solutions. This may increase the exploitation capability and decrease the diversity of search space. An improved migration an updated feature appears in poor solution. Where updated features coming from our proposed migration operator. We used one scaling function for maintaining the exploration (diversity) and exploitation capability. But we have to use this scaling function in proper way to maintaining diversity and exploitation capability. If F =0 it is similar to traditional BBO. Hence if F =0 diversity of search space is decreasing and exploitation capability decreasing. If F =1 diversity of search space is increasing and exploitation capability is increasing. For maintaining these two things we have taken F =0.5. To analyze the effect of this proposed operator on the algorithms performance, we have designed five set of experiments. In this set, GAPAM, VDGA, BBO, MOMSA and Improved BBO were run. We measure the fitness of each habitat according to fitness function which is given in section 3.2. We have used 8 Balibse datasets for these experiments (4 from each of reference set 1 and 2) which is illustrated in Fig. 6).

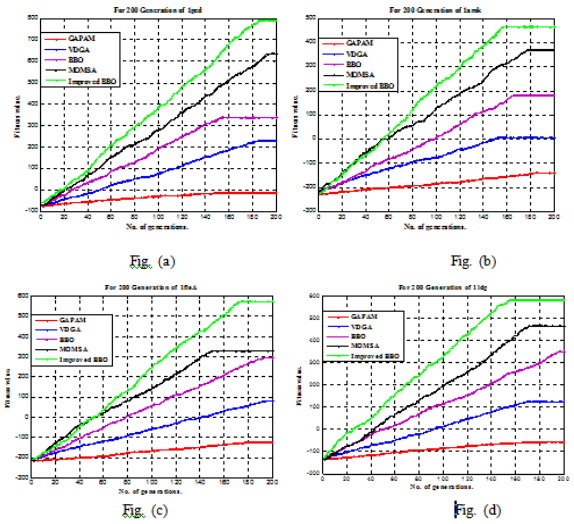

Fig. 6. Performance of Improved BBO and some *existing* methods per generation w.r.to reference

5.2 Experimental Results and Analysis

(1) Comparison of IBBOMSA with MOMSA ,VDGA and GAPAM

In order to examine the performance of our proposed method IBBOMSA we compare with well-known existence methods such as VDGA[28], GAPAM[27] and MOMSA[29] which is best methods for multiple sequence alignment in recent time. We have taken selected dataset from MOMSA for comparing our proposed method to other methods in appropriate manner. The authors chose 56 test cases in Bali base 2.0, which contains 18 test cases from Reference 1, 23 test cases from Reference 2, 11 test cases from Reference 3, and two test cases from Reference 4 and 5 respectively. As mentioned in section 3.3 for calculating fitness function of multiple sequence alignment. We calculate fitness value of corresponding multiple sequence alignment is recorded. IBBOMSA is performed for 10 times and the best of their results are recorded. Fig. 7,8,9,10,and 11 show the results of IBBOMSA ,MOMSA, VDGA, and GAPAM on Bali base reference 1, 2,3 4 and 5 respectively

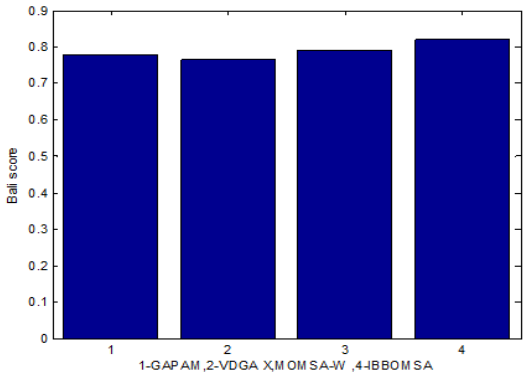

Fig 7. The result of IBBOMSA, MOMSA-W, VDGA and GAPAM on Bali base Reference 1

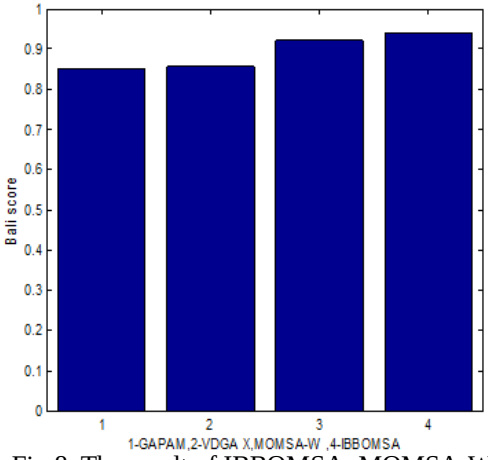

Fig 8. The result of IBBOMSA, MOMSA-W, VDGA and GAPAM on Bali base Reference 2

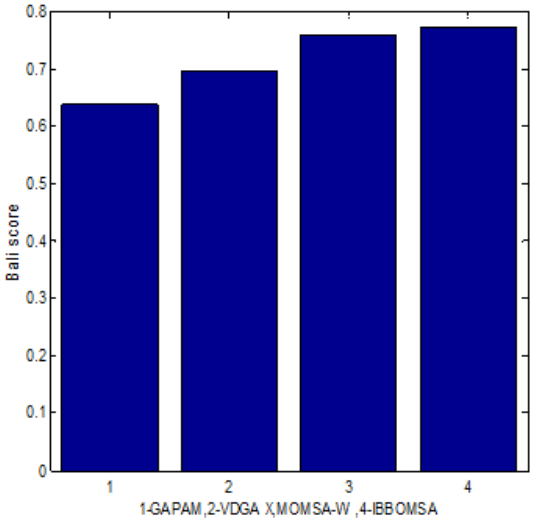

Fig 9. The result of IBBOMSA, MOMSA-W, VDGA and GAPAM on Bali base Reference 3

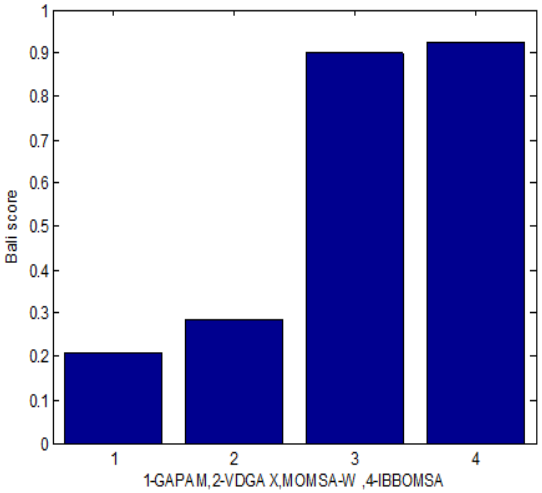

Fig 10. The result of IBBOMSA, MOMSA-W, VDGA and GAPAM on Bali base Reference 4

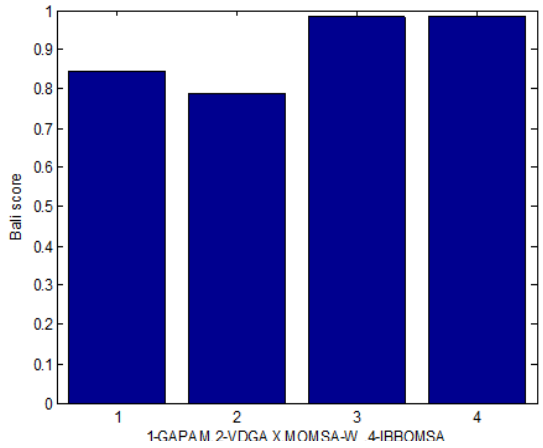

Fig 11. The result of IBBOMSA, MOMSA-W, VDGA and GAPAM on Bali base Reference 5

6 Conclusions

In this paper, we have proposed an improved BBO algorithm for solving Multiple Sequence Alignment. We design a new migration operator to maintaining exploration and exploitation. However, we have to use scaling function carefully. We compared the new algorithm with the existing BBO algorithm. It shows that new algorithm is superior to existing BBO or at least competitive. To test our present approach, we considered a good number of benchmark datasets from Bali base 2.0, so as to cover all the test sets of MOMSA. Therefore, the corresponding Bali score of this solution was used to compare with other methods, as they used Bali score as their measure of the quality/accuracy of the MSA. The experimental results proved that proposed BBO performed better for most of the test cases. Even the solution of proposed BBO was not always the best for some test cases it was always close to the best. The proposed method performed better than the others because of its improved migration operator to help maintain diversity of search space. After the experimental analysis, we can say that the proposed method

References

[1] Bahr, A., Thompson, J.D., Thierry, J.C., (2001): Poch, O; Balibase (benchmark alignment database): enhancements for repeats, transmembrane sequences and circular permutations. Nucleic Acids Research. 29,323-326.

[2] Bonizzoni, P., Della Vedova, G., (2001): The complexity of multiple sequence alignment with sp-score that is a metric. Theoretical Computer Science.259,63-79.

[3] Cai, L., Juedes, D., Liakhovitch, E.: Evolutionary computation techniques for multiple sequence alignment. In, (2000): Evolutionary Computation Proceedings of the Congress.829-835.

[4] Carrillo, H., Lipman, D., (1988) : The multiple sequence alignment problem in biology. SIAM Journal on Applied Mathematics.48, 1073-1082.

[5] Chellapilla, K., Fogel, G.B., (1999): Multiple sequence alignment using evolutionary programming. In: Evolutionary Computation CEC Proceedings of the Congress.

[6] Dayhoff, M.O., Schwartz, R.M., (1978) : A model of evolutionary change in proteins. In: In Atlas of protein sequence and structure.

[7] Eddy, S.R.:, (1995) Multiple alignment using hidden markov models.3,114-120.

[8] Feng, D., Johnson, M., Doolittle, R, (1985) .: Aligning amino acid sequences: comparison of commonly used methods .Journal of Molecular Evolution 21: 112-125.

[9] Feng, D.F., Doolittle, R.F., (1987): Progressive sequence alignment as a prerequisitetto correct phylogenetic trees. Journal of molecular evolution.25,351-360.

[10] Gondro, C., Kinghorn, B., (2007): A simple genetic algorithm for multiple sequence alignment. Genetics and Molecular Research.6,964-982.

[11] Gusfield, D., (1997): Algorithms on strings, trees and sequences computer science.

[12] Horng, J.T., Lin, C.M., Liu. B.J., Kao, C.Y., (2000) ;Using genetic algorithms to solve multiple sequence alignments. In: GECCO.

[13] Ishikawa, M., Toya, T., Totoki, Y., Konagaya, A. (1993): Parallel iterative aligner with genetic algorithm. Genome Informatics.4,84-93.

[14] Kim, J., Pramanik, S., Chung, M.J., (1994): Multiple sequence alignment using simulated annealing. Computer applications in the biosciences: CABIOS.10,419-426.

[15] Lee, Z.J., Su, S.F., Chuang, C.C., Liu, K.H., (2008): Genetic algorithm with ant colony optimization (ga-aco) for multiple sequence alignment. Applied Soft Computing.8,55-78.

[16] Lukashin, A.V., Engelbrecht, J., Brunak, S., (1992):Multiple alignment using simulated annealing: branch point definition in human mrna splicing. Nucleic acids research.20,2511-2516.

[17] Needleman, S.B., Wunsch, C.D. (1970): A general method applicable to the search for similarities in the amino acid sequence of two proteins. Journal of molecular biology.48,443-453.

[18] Notredame, C., Higgins, D.G., (1996): Saga: sequence alignment by genetic algorithm. Nucleic acids research.24, 1515-1524.

[19] Simon, D.: Biogeography-based optimization. IEEE Trans Evol Comput.12,702–713(2008)

[20] Shyu, C., Sheneman, L., (2004) : Foster, J.A; Multiple sequence alignment with evolutionary computation. Genetic Programming and Evolvable Machines.5,121-144.

[21] Taheri, J., Zomaya, A.Y. ,(2009): Rbt-ga: a novel metaheuristic for solving the multiple sequence alignment problem. Bmc Genomics.10.

[22] Taheri, J., Zomaya, A.Y., (2010): Rbt-l: A location based approach for solving the multiple sequence alignment problem. International journal of bioinformatics research and applications.6,37-57.

[23] Taheri, J., Zomaya, A.Y., Zhou, B.B. .(2008)

[24] : RBT-L: A Location Based Approach for Solving the Multiple Sequence Alignment Problem. School of Information Technologies, University of Sydney.

[25] Taylor, W.R., (1988) : A flexible method to align large numbers of biological sequences. Journal of Molecular Evolution.28,161-169.

[26] Thompson, J.D., Higgins, D.G., Gibson, T.J.: Clustal w, (1994): improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. Nucleic acids research.22,4673-4680.

[27] Thompson, J.D., (1999): Plewniak, F., Poch, O.: Balibase: a benchmark alignment database for the evaluation of multiple alignment programs. Bioinformatics.15,87-88.

[28] Naznin, F., Sarker, R., Essam, D., (2012) : Progressive Alignment Method Using Genetic Algorithm for Multiple Sequence Alignment. IEEE Transaction on Evolutionary Computation.16,615-631.

[29] Naznin, F., Sarker, R., and Essam, D., (2011): Vertical decomposition with Genetic Algorithm for Multiple Sequence Alignment. BMC Bioinformatics.12,353.

[30] Zhu, H., He, Z., and Jia, Y., (2015): A Novel Approach to Multiple Sequence Alignment Using Multi-objective Evolutionary Algorithm Based on Decomposition. IEEE Journal of Biomedical and Health Informatics. 1-11.