Multiple Sequence Alignment by Differential Evolutionary Algorithm with New Mutant

Lakshmi Naga Jayaprada.Gavarraju and K. Karteeka Pavan

Abstract: ***Multiple Sequence Alignment (MSA) is vital in Bioinformatics, helps in finding evolutionary relationships among multiple species. MSA is a NP-complete problem. Though there are a number of tools recent Meta-heuristics are found to be effective in solving MSA problem. Differential Evolutionary Algorithm (DE) is one of the optimization algorithms with various mutants. This work proposes a new mutant for DE, defined using local best and worst chromosomes with current generation population. The performance of the new mutant is evaluated using 50 well known bench mark data sets in sabre (SABMARK v1.65). The results are matched with all the other DE mutants, Genetic Algorithm (GA) and recent Teacher Learner Based Optimization algorithm (TLBO). The proposed DE mutant outperformed all the other DE mutants, GA and TLBO in solving MSA problem.***

Keywords : MSA, GA, DE, TLBO.

I. INTRODUCTION

MSA is a method of aligning multiple sequences with an aim to maximize the match score. Many computational algorithms were developed as solutions to MSA problem. But almost all the solutions are found to be computationally complex. Many heuristic algorithms were also developed for an optimal alignment among multiple sequences. Dynamic programming approach is used to find optimal alignment for MSA and is proved as NP-complete problem [1]. Carrillo and Lipman have defined a solution based on computing each pairwise alignment, which leads to multiple sequence alignment and found to be a much time consuming process. The success of Heuristic search optimization methods in many applications are encouraged many researchers to develop new optimization algorithms for the problem of MSA [2].

Progressive alignment algorithm is the widely used heuristic approach [3] developed by Feng and Doolittle. It is a guide tree approach with two phases. Guide tree is to be constructed using a clustering method in the first phase. Second phase finds MSA using the guide tree. The progressive alignment methods are very efficient in finding alignment on even hundreds to thousands of sequences. However, these are sensitive to initial values. Most familiar progressive alignment methods are from CLUSTAL family [4][5]. One of the Progressive alignment methods CLUSTAL OMG is used for protein sequences and MAFFT is used for DNA sequences. T-Coffee[6] is another method that produces more accurate alignment compared to CLUSTALW. But it takes more computational time. PSAlign [7] is a semi progressive algorithm which produces quality alignment than its previous algorithms. Further iterative alignments are developed using progressive methods for Sequence Alignment problem. Most familiar iterative approaches are DALIGN [8] and MUSCLE [9].

Evolutionary algorithms are proven to be efficient heuristic search optimization methods from a decade for many applications [10]. These nature inspired algorithms are evolved [11] constructed on the Darwinian opinion only "fittest will win". Holland [12] first proposed the elementary principles of GA. It is motivated by natural selection process where tougher individuals can remained in the competing environment. In each round of the progressing process a successive generation is produced from the present population by mixing the genes of parents [13]. Crossover and mutation were the essential operators in GA and selection criterion is taken as another operator.

DE is developed as a real parameter optimization technique in evolutionary computing [14]. In 1995 Storn and Price were first published the DE algorithm [15]. It follows the similar steps of GA. Mutation is in all the stages. Mutation is an unexpected alteration in the gene characteristics of a chromosome. Many mutation strategies proposed for DE. In DE, for each iteration the population becomes better in fitness status, but never depreciates. Core control parameters related to the DE are the mutation factor (F), the crossover rate (Cr) and the population size (NP). Many studies specify that DE is simple and straightforward when associated to other algorithms in evolutionary category and shows better performance. The performance of DE with respect to accuracy moulds it suitable for several real-time optimization applications.

Almost all these algorithms are sensitive to algorithmic specific control parameters. In 2011, Rao et al. proposed TLBO [16]. It won't take any algorithmic specific parameters and it has two phases: teacher phase and learner phase. Because it is having two phases in its algorithm, when applied on multiple sequence alignment problem it has shown better performance than GA and DE with all its mutants. But is observed that it required more time compared to GA and DE. GA results in a poor performance among all the three methods GA, DE and TLBO. This motivated us to find a new mutant to DE which works better than TLBO in comparatively reduced time.

This work proposes a novel mutant for DE considering best and worst fit candidate solutions from the current population. This mutant is compared with all the other five established DE mutants like "DErand1", "DEbest1", "DErand2", "D/best2",

Revised Manuscript Received on November 19, 2019

Lakshmi Naga Jayaprada. Gavarraju, Assoc.Prof, Dept. of Computer Science & Engineering, Narasaraopeta Engineering College [Autonomous], Narasaraopet, Guntur(Dt), A.P., India.

Kanadam Karteeka Pavan, Professor & Head Department of Computer Applications, R.V.R.& J.C.College of Engineering [Autonomous], Chowdavaram , Guntur , A.P., India.

"DEcurrent-to-best1", GA and TLBO. The proposed method has shown the significant improvement in performance over all the other DE mutants, GA and TLBO. To accomplish testing on proposed mutant, bench mark datasets from "sabre" are taken into account. The work is present in 6 parts. Part 1 is introduction. Part2 discusses MSA problem and Objective function. Part 3 focuses on Differential Evolutionary Algorithm and the proposed new mutant. Experimental Outcomes and test data sets are furnished in Part4. Part 5 is limitations and finally Part 6 consists of conclusions and future endeavours.

II. MSA PROBLEM AND OBJECTIVE FUNCTION

MSA is computationally intense problem and validating the alignments is also critical. It takes so much time for computations and to align them. Selection of fitness functions in evolutionary algorithms is primitive. The following subsection covers MSA and objective functions.

A. Demonstration of MSA with example set of sequences

Usually sequences are aligned by introducing gaps at random positions with an aim to maximize the matches. The following are the details of some example test sequences and their alignment.

Figure 1 shows test sequences of data set "sabre" (SABMARK v1.65) known as sup-360.

```
>d1en2a2
DHRCGAAVGNPPCGQDRCCSVHGWCGGGNDYCSGSKCQYRC
>d1hev__
EQCGRQAGGKLCPNNLCCSQWGWCGSTDEYCSPDHNCQSNCKD
>d1mmc__
VGECVRGRCPSGMCCSQFGYCGKGPKYCGR
>d9wgaa2
ATCPNNHCCSQYGHCGFGAEYCGAGCQGGPCRAD
```

Figure 1: Test Sequences

III. REVIEW CRITERIA

The input sequences are aligned using gaps at random positions. Figure 2 is the snapshot of alignment.

```
DH-RCGAAVG-NP-PCGQD-RCC-SVH-GWC--GGGNDYC-SGSKC-QY-RC
--EQCGRQAGGKLCPNNLCC--SQWGWCGS-TDEYCSPDHNCQSNC-K-D--
-V--GEC-VRGR---CPS---GMCCSQFG-Y-C--GKGP-K-YC-----G-R
-A-TCP-NNH--C--CSQ-YGHCGF-G-AE-YCG-AGCQG--GPCR---A-D
```

Figure 2: Aligned Sequences

Sequences can be aligned in many ways with the choice of objective functions. Various objective functions are presented in the following section.

B. Objective Function

Alignment quality can be measured using various objective functions. Numerous Objective Functions were developed over the period to calculate the fitness value of each and every alignment within the generated population. They are Sum-of-Pairs Score (SoPs), Weighted Sum-of-Pairs Score (WSPs) [5], Coffee [17], T-Coffee [6] etc. In this work Sum-of-Pairs Score is preferred. SoPs is easy to use for assessing the alignment. Because we use SoP as an Objective function, its process is explained in detail.

Assume that there are P sequences in alignment each of which contains Q columns. And the elements of one column are designated as M11,M21,M31….,MP1.

$$SoP = \sum_{a=1}^{Q} \sum_{b=1}^{P-1} \sum_{c=b+1}^{P} Score(Mba, Mca) \quad (1)$$

where

$$Score(Mba,Mca) = \begin{cases} match\ if\ Mba = Mca\ and\ Mba \neq '-'\ and\ Mca \neq '-' \\ gap\ if\ Mba = '-'\ or\ Mca = '-' \\ mismatch\ if\ Mba \neq Mca \end{cases} \quad (2)$$

Where `-' represents a gap and is allocated a mark of -2, match is allocated a mark of '+20' and mismatch is allocated the mark of '-1'. Match is assigned such a high value '+20' in order to get positive fitness score. The suggested technique is explained in the next part.

III. DIFFERENTIAL EVOLUTION AND PROPOSED NEW MUTANT

With the motivation of finding a new mutant to DE which gives better performance than TLBO, this paper proposes a new mutant for DE Algorithm and it is known as "DE/optimal". Nearly two decades ago in 1995 Storn and Price first published their Evolutionary algorithm known as DE algorithm [13]. DE is very simple when related with all other Evolutionary Algorithms. When matched with other evolutionary algorithms DE is also having less control parameters like Cr, F and NP [14]. The important steps of DE are as shown in Figure 3.

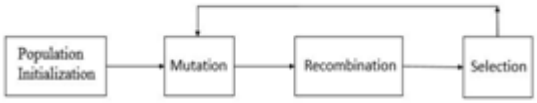


Fig 3: Differential Evolutionary Algorithm

A. Differential Evolutionary Algorithm

1. Initialize control parameters population size (NP), mutation factor (F) and cross-over rate (Cr).
2. Initialize population depending on population size.
3. Repeat the subsequent steps till the specified number of generations is reached.

   For i=1 to s
4. Find the fitness of each candidate solution using Sum of pairs score.
5. Mutation: Find the donor vector $V_{i,G} = X_{r1,G} + F(X_{r2,GN} - X_{r3,GN})$.

   where "Xr1,G", "Xr2,G" & "Xr3,G are the three random candidate solutions, Vi,G is the newly generated candidate solution and F is a random number between 0 and 1.
6. Crossover: Develop a trial vector Ui,G for the ith target vector Xi,G by using binomial crossover.
7. Selection: Evaluate trial vector Ui,G

   if f (Ui,G) ≤ f (Xi,G), then Xi,G+1 = Ui,G

   else Xi,G+1 = Xi,G

   end if

   end for

For the mutation operation various mutation alternatives are utilized. Already available mutants are shown below.

DErand1:$V_{i,G} = X_{p1,G} + F(X_{p2,G} - X_{p3,G})$
DEbest1: $V_{i,G} = X_{b,G} + F(X_{p1,G} - X_{p2,G})$
DEcurrent-to-best1:$V_{i,G}=X_{i,G}+F(X_{b,G} - X_{i,G})+F(X_{p1,G} - X_{p2,G})$
DEbest2: $V_{i,G} = X_{b,G} + F(X_{p1,G} - X_{p2,G})+F(X_{p3,G} - X_{p4,G})$
DErand2": $V_{i,G} = X_{p1,G} + SF(X_{p2,G} - X_{p3,G})+F(X_{p4,G} - X_{p5,G})$
Where p1, p2, p3, p4, p5 ε [1, …. , NP] are randomly chosen integers, and p1 ≠ p2 ≠ p3 ≠ p4 ≠ 5 ≠ i. F is the mutation factor controlling the amplification of the differential evolution. $X_{b,G}$ is the best individual vector with the best fitness value in the population at generation *G*.

B. Proposed DE Mutant (Modified DE)

Newly proposed mutant for DE is as follows.

$$\mathbf{V_{i,G} = X_{b,G} + F(X_{i,G} - X_{w,G})}$$

$X_{b,G}$ is best fit candidate solution, $X_{w,G}$ is worst fit candidate solution, Vi,G is the newly generated candidate solution, F is mutation factor and is a random number between 0 and 1.

IV. EXPERIMENTAL OUTCOMES AND TEST DATA SETS

In this section the new proposed DE mutant is compared with all the other well established five DE mutants like "DErand1", "DEbest1", "DErand2", "DEbest2", "DEcurrent-to-best1", GA and TLBO. The newly proposed DE mutant outperformed all the above said algorithms. In this paper, DE algorithm with new mutant is coded in MATLAB and executed on the PC with Intel® core(TM) i3-6100 CPU @ 3.70 GHZ with 4GB RAM under Windows 7.1 Platform.

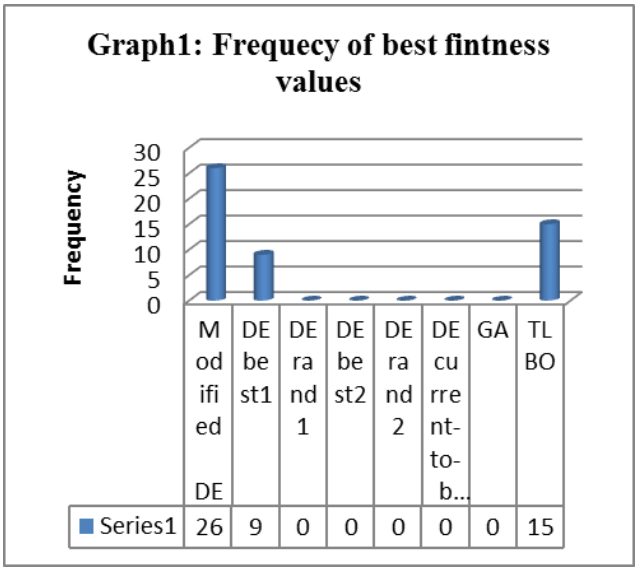
Experiments are conducted to evaluate the performance of the proposed method using the bench mark dataset "sabre" (SABMARK v1.65). 50 test data sets are taken from "sabre" and the results are tabulated in Table 1.

A. Discussions

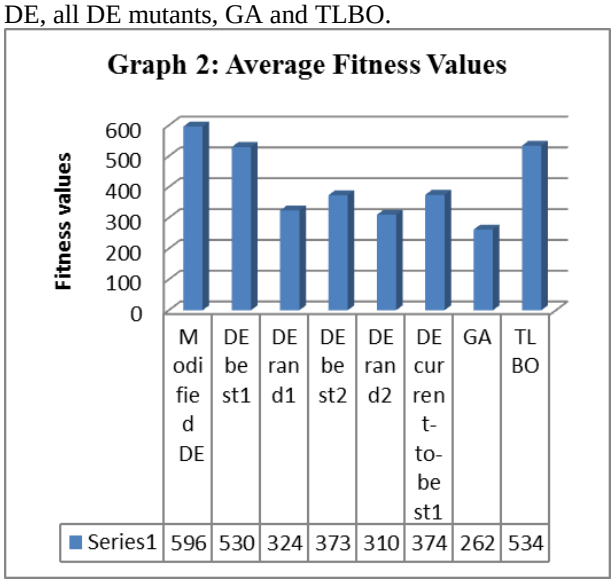
The proposed DE mutant is applied on 50 test datasets taken from well-known bench mark datasets "sabre". Among the 50 test datasets for 27 datasets the proposed DE mutant has shown the better performance. "DEbest1" has given better performance for 8 test data sets. TLBO has demonstrated best performance in 15 test data sets. Although DE/best1 and TLBO is showing the better performance for some test datasets the time required to run the algorithm when using new mutant is very less when compared with DEbest1 and TLBO. Frequency of the best fitness values of all algorithms is as shown in Table 2 and depicted in Graph 1.

Table 2: Frequency of best fitness values

Modified DE	DE best1	DE rand1	DE best2	DE rand2	DE current-to-best1	GA	TLBO
26	9	0	0	0	0	0	15



Graph 2 depicts the average fitness values of the Modified DE, all DE mutants, GA and TLBO.

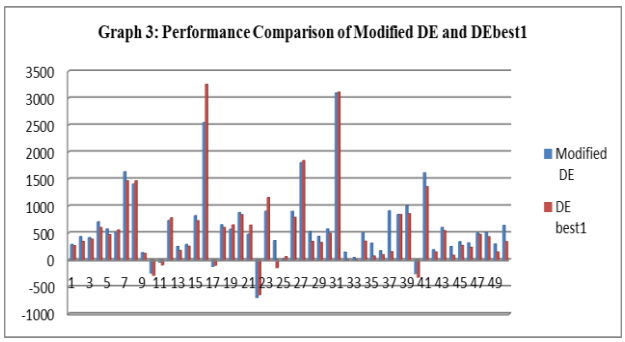


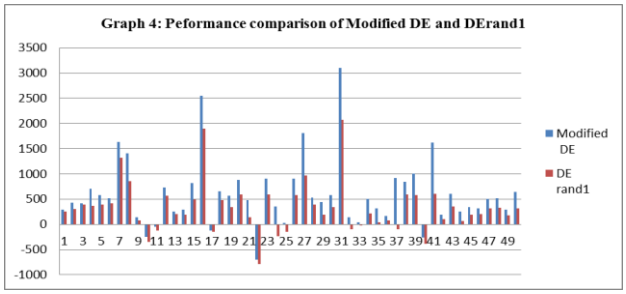
Graph 3 depicts the performance comparison of Modified DE with DEbest1. Graph 4 depicts the performance comparison of Modified DE with DErand1. Graph 5 depicts the performance comparison of Modified DE
with DEbest2. Graph 6 depicts the performance comparison of Modified DE with DErand2. Graph 7 depicts the performance comparison of Modified DE with DEcurrent-to-best1. Graph 8 depicts the performance comparison of Modified DE with GA. Graph 9 depicts the performance comparison of Modified DE with TLBO.

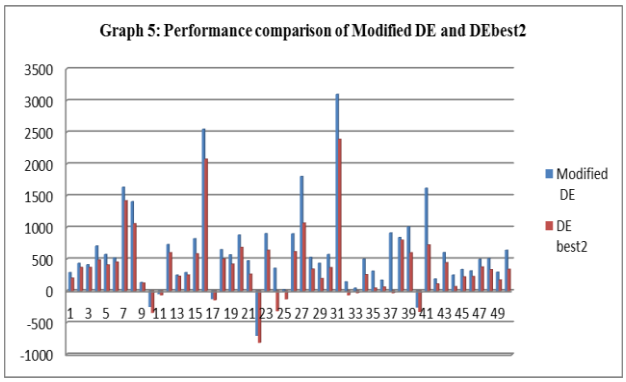
Table 1: Modified DE, All DE mutants, GA and TLBO Performance Comparison for MSA

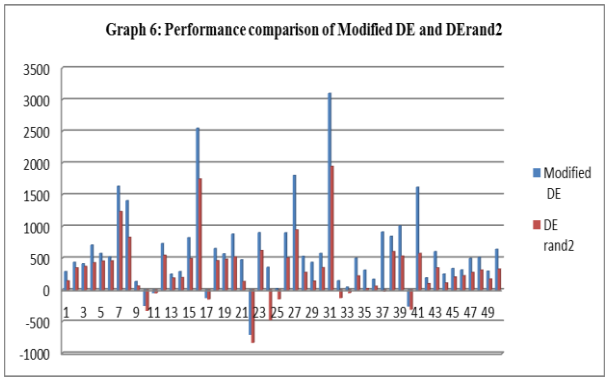
S. No.	Name of the sequences	No. of Sequences	Modified DE	DE best1	DE rand1	DE best2	DE rand2	DE current-to-best1	GA	TLBO
1	sup-009	3	**289**	267	254	203	144	206	104	222
2	sup-017	3	**435**	343	306	370	349	433	245	416
3	sup-019	3	**413**	388	391	370	370	396	264	355
4	sup-030	3	**708**	602	361	488	428	534	260	447

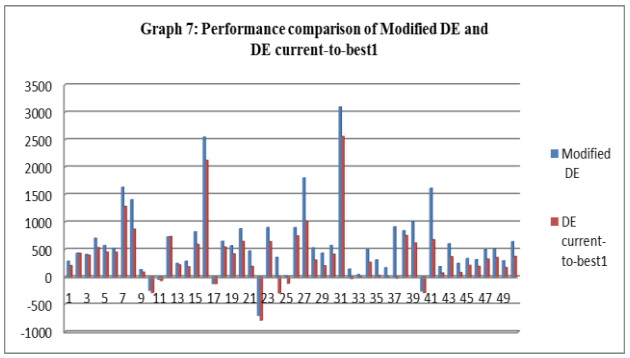
5	sup-031	3	**575**	470	390	411	455	453	309	456
6	sup-300	3	522	**556**	411	456	457	449	404	390
7	sup-024	3	**1635**	1467	1318	1422	1235	1286	1324	1491
8	sup-039	4	1408	**1470**	852	1060	830	874	667	1098
9	sup-040	3	134	119	82	125	62	89	100	**180**
10	sup-041	4	-250	-292	-349	-344	-327	-292	-241	**522**
11	sup-043	3	-49	-98	-120	-69	-51	-73	-95	**126**
12	sup-050	3	732	**781**	571	603	545	737	606	447
13	sup-054	3	250	176	198	230	189	220	178	**604**
14	sup-080	3	**290**	252	191	251	194	187	249	217
15	sup-082	4	**823**	726	496	585	494	595	434	645
16	sup-083	4	2551	**3261**	1899	2077	1751	2123	1279	1686
17	sup-107	3	-127	-107	-149	-146	-149	-131	-143	**126**
18	sup-111	3	**652**	603	479	504	462	544	395	557
19	sup-124	3	570	**650**	340	422	483	421	483	273
20	sup-127	3	**881**	841	586	688	520	646	436	584
21	sup-139	5	475	647	142	264	134	194	113	**1038**
22	sup-002	4	-705	-654	-793	-817	-833	-793.6	-679	**-641**
23	sup-008	3	903	**1157**	587.6	641	620	642.2	517	612
24	sup-034	7	**358**	-149	-239	-318	-475.4	-299	-419	222
25	sup-094	3	21	**60**	-148	-131	-146	-122.8	-103	-41
26	sup-312	3	**901**	791	579.8	618	503.9	747.8	331	678
27	sup-306	4	1806	**1843**	971	1069	947	1016.6	873	1004
28	sup-304	3	**531**	342	386	345	275.8	309.6	343	395
29	sup-423	4	**437**	323	194	197	138	204	180	348
30	sup-417	3	**575**	492	345	368	350	416	405	367
31	sup-416	5	3100	**3114**	2078	2392	1948	2562	993	1465
32	sup-258	5	145	-19	-101	-68	-126	-38	-196	**733**
33	sup-358	3	45	-10	-19.2	-34.4	-49.6	-8.2	-97	**105**
34	sup-360	4	**502**	347.2	219	256	221.2	266.4	274	435
35	sup-364	4	312	71.8	39.4	46	22.8	28.8	100	**369**
36	sup-370	3	**170**	95.6	82.4	62	58.4	-0.2	22	130
37	sup-371	9	913	151.4	-93.6	-35.4	-24.4	-29.8	287	**2726**
38	sup-372	3	**844**	840	592.2	802	605.8	755.2	479	677
39	sup-375	3	**1011**	856.6	580.4	601	533.4	619.8	441	571
40	sup-379	5	-261	-326	-390	-329	-310.4	-292.8	-410	**370**
41	sup-381	6	**1619**	1359	608.6	724	575.8	679	501	1075
42	sup-386	3	190	149.6	99	112	98.8	72.4	80	**279**
43	sup-387	3	**604**	545.2	348.8	444	348	369	292	334
44	sup-393	4	**250**	83.8	65.6	70	110.8	81	14	180
45	sup-398	3	**338**	267.8	186.4	223	203	212.2	147	250
46	sup-399	3	315	235.4	207.6	228	223	193	159	**646**
47	sup-401	4	**498**	472.8	318.6	378	275.8	327.2	495	471
48	sup-403	3	**512**	428.2	326.8	336	312.4	354.8	325	346
49	sup-406	3	296	147.8	183.6	172	171	173.6	136	**333**
50	sup-417	3	**642**	339.2	317.6	341	329.4	378	222	368
	Average fitness		**595.78**	529.6	323.6	373	309.65	374.304	261.7	533.7

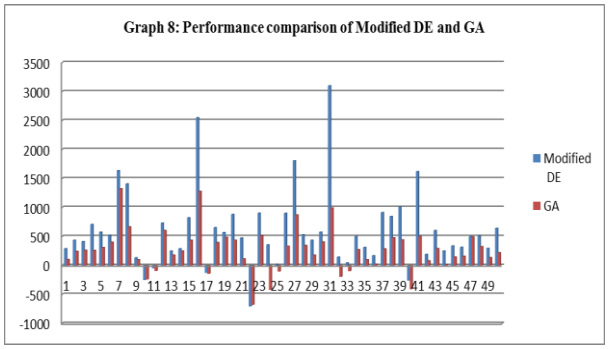


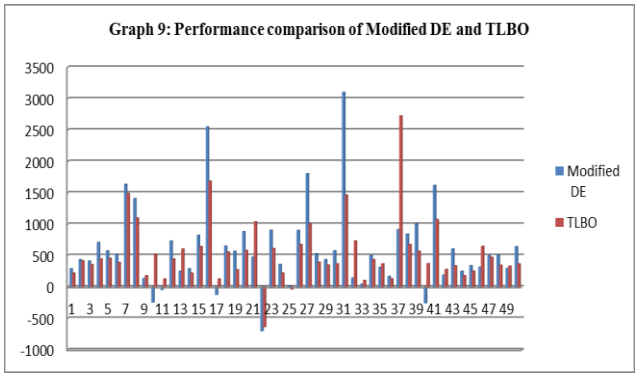












V. LIMITATIONS

To test the proposed DE mutant a maximum of 9 test sequences with average length of sequence as 200 characters are only considered. Increase in length of test sequences or increase in number of test sequences definitely increases the execution times.

VI. CONCLUSION AND FUTURE ENDEAVOURS

MSA is the task of aligning multiple sequences of type protein or DNA to find the evolutionary relationships among multiple sequences. While aligning if we get maximum fitness score we can say that the sequences are more evolutionarily related**.** DE is actually a stochastic real parameter optimization based technique which requires more time to run.

This paper proposes a new DE mutant which uses best and worst candidate solutions to find the new offspring. Tests are conducted on sabre bench mark datasets. Sum-of-pairs objective function is used to calculate the fitness value. Tests can also be conducted with other objective function like column score.

The proposed mutant outperformed all the DE mutants, GA and TLBO. It was also found that the proposed mutant is also consuming less time compared with all other DE mutants and TLBO. In order to further enhance the execution speed parallelization can be applied on all the algorithms and also can go for distributed computing.

REFERENCES:

1. Wang L, Jiang T (1994), "On the complexity of multiple sequence alignment", J Comput Biol. 1 (4): 337–348. doi:10.1089/cmb.1994.1.337.
2. Lipman DJ et.al. (1989), "A tool for multiple sequence alignment", Proc Natl Acad Sci U S A. 86 (12): 4412–4415. doi:10.1073/pnas.86.12.4412.
3. Feng DF, Doolittle RF (1987), "Progressive sequence alignment as a prerequisite to correct phylogenetic trees", J Mol Evol. 25 (4): 351–360.
4. Higgins DG, Sharp PM (1988), "CLUSTAL: a package for performing multiple sequence alignment on a microcomputer", Gene. 73 (1): 237–244, doi: 10.1016/0378-1119(88)90330-7.
5. Thompson JD et.al., (1994), "CLUSTAL W: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice", Nucleic Acids Res. 22 (22): 4673–4680, doi: 10.1093/nar/22.22.4673.
6. Notredame C et.al., (2000), "T-Coffee: A novel method for fast and accurate multiple sequence alignment", J Mol Biol. 302 (1): 205–217. doi:10.1006/jmbi.2000.4042.

7. Sze SH, Lu Y, Yang Q (2006), "A polynomial time solvable formulation of multiple sequence alignment". J Comput Biol. 13 (2): 309 319. doi:10.1089/cmb.2006.13.309.
8. Brudno M et.al., (2003), "Fast and sensitive multiple alignment of large genomic sequences", BMC Bioinformatics. 4: 66, doi:10.1186/1471-2105-4-66.
9. Edgar RC (2004), "MUSCLE: multiple sequence alignment with high accuracy and high throughput". Nucleic Acids Research. 32 (5): 1792–97. doi:10.1093/nar/gkh340.
10. Huazheng Zhu et.al., "A Novel Approach to Multiple Sequence Alignment Using Multi objective Evolutionary Algorithm Based on Decomposition, IEEE Journal of Biomedical and Health Informatics, volume 20 issue 2.
11. K. F. Man, Member et.al., IEEE,(1996) "Genetic Algorithms: Concepts and Applications", IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, VOL. 43, NO. 5, OCTOBER 1996.
12. J. H. Holland (1975), "Adaption in Natural and Artificial Systems", Cambridge, MA: MIT Press.
13. R. Storn and K. V. Price, (1995), "Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces", ICSI, USA, Tech. Rep. TR-95-012.
14. Swagatam Das, Member, IEEE, and Ponnuthurai Nagaratnam Suganthan, Senior Member, IEEE, (2011), "Differential Evolution: A Survey of the State-of-the-Art", IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION, VOL. 15, NO. 1, FEBRUARY 2011.
15. R. Storn and K. V. Price, (1995), "Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces", ICSI, USA, Tech. Rep. TR-95-012.
16. R.V. Rao et al., (2016), "Teaching Learning Based Optimization Algorithm",© Springer International Publishing Switzerland 2016, doi:10.1007/978-3-319-22732-0_2.
17. C. Notredame et.al., (1998), "COFFEE: An objective function for multiple sequence alignments", Bioinformatics, vol. 14, no. 5, pp. 407–422.