An improved chemical reaction-based approach for multiple sequence alignment

Rohit Kumar Yadav* and Haider Banka
Department of Computer Science and Engineering, Indian Institute of Technology (Indian School of Mines), Dhanbad 826 004, India

In bioinformatics, multiple sequence alignment (MSA) is an NP-hard problem. Nature-inspired approaches can provide an approximate solution compared to conventional approaches. In this article, the MSA problem is dealt with using chemical reaction optimization (CRO). The limitations of CRO are slow convergence and low population diversity. Therefore, the initialization process is improved by pairwise alignment technique which maintains diversity. In the performance analysis, we have taken benchmark datasets from Bali base version 2.0. The Bali score of the proposed approach is compared with those of the existing approaches such as SB-PIMA, SAGA, RBT-GA and GAPAM, HMMT. Simulation results confirm the superiority of the proposed approach over others.

Keywords: Bioinformatics, chemical reaction optimization, multiple sequence alignment, population diversity.

Multiple sequence alignment (MSA) is an alignment problem where more than three amino acids or protein sequence participate in the alignment. We can solve many biological problems with the help of MSA. It is useful to suggest primary and secondary structures of RNAs and proteins. MSA can reconstruct phylogenetic tree. We can also predict the function of unknown amino acid sequences using the phylogenetic tree. MSA can find similarity between sequences, which is helpful to know the function and structure of similar protein or amino acid sequences. However, MSA maximizes the matching component as well as minimizes the mismatching component, which is problematic. There are many problems in bioinformatics such as finding ancestral and hereditary relationships, which can be solved by MSA. Hence, the MSA problem must be dealt with in the efficient manner.

In 1970, Needleman and Wunsch proposed an algorithm using dynamic programming (DP) to solve the MSA problem. This algorithm is useful to solve pairwise sequence alignment problem. DP can also solve the MSA problem and give an optimal solution. However, the process is time-consuming, especially when the number and length of sequences are large. Hence the MSA problem becomes NP-hard and it is not feasible to use DP to solve the MSA problem. We need to develop new algorithms for the same.

The MSA problem can be solved in a systematic way by progressive alignment. This approach is less complex in terms of time and space for solving the MSA problem. This method has used repeatedly Needleman and Wunsch algorithm to find guide tree between MSA. The progressive alignment method initially aligns more similar sequences, after which it incrementally aligns more dissimilar sequences or groups of sequences in the initial alignment. CLUSTAL W is the standard representation of the progressive method. In the first step, we assign weights to every pair of sequences in a partial alignment. We assign small weights for the most similar sequences and big weights for most divergent sequences. Next, we take a substitution matrix which defines the score between two residues of a protein sequence based on similarity. Two types of gaps are introduced in the third step. The first is residue-specific gap and the second is locally residue gap penalties. These steps are integrated into CLUSTAL W, which is freely available. Progressive alignment method performs better for MSA package in terms of accuracy and time. However, this method has some limitations – dependency on initial alignment and choice of scoring scheme. In other words, we need to align more similar sequences in the initial stage. If not, the solution may be trapped in local optima.

An iterative method initially starts with a random solution and improves the solution in an iterative manner until no more upgradation is possible. In this case, the result does not depend on the initial population. The main aim of this method is to find the global optimum. In order to solve the MSA problem, the objective of the iterative method is to find an alignment which is the globally optimal alignment. Simulated annealing is an example of the iterative process. Hidden Markov Model Training (HMMT) method is based on simulated annealing process. The drawback of the iterative method is that the solution may be trapped in local optima. So researchers have switched to another method for solving the MSA problem, which is evolutionary or nature-inspired method.

The evolutionary method starts with a random initial population. In the second step, we calculate fitness value of each solution using an objective function. In the third step, we modify the initial solution using some operators and continuously use this operator until we reach the global optimum. In this method, the initial solution is

*For correspondence. (e-mail: rohit.ism.123@gmail.com)
originated in a random way, after which we apply evolutionary operators to enhance the similarity of MSA. Some algorithms available are based on evolutionary computations for MSA\textsuperscript{10-14}, while others are based on genetic algorithms for MSA such as SAGA\textsuperscript{15}, MSA-GA\textsuperscript{15}, RBT-GA\textsuperscript{16} and genetic algorithm with progressive alignment method (GAPAM)\textsuperscript{17}. In the case of GAPAM, Naznin et al.\textsuperscript{17} have taken 56 different types of datasets from reference sets 1–5. The limitation of these evolutionary-based algorithms is that the result may be trapped in local optima. An improved chemical reaction optimization (ICROMSA) has been proposed to solve the MSA problem. We have developed a technique to generate initial molecular structure which is helpful to converge to global optimal alignment. We have also compared the classical chemical reaction optimization (CRO) and ICROMSA with respect to some Bali base datasets and found that the latter can perform better in most cases.

**Basics of CRO**

Most of the swarm intelligence techniques developed earlier are based on constant population, but CRO is based on variable length population\textsuperscript{18}. In CRO, a solution is represented by the structure of a molecule. A molecule has two types of energy – potential energy (PE) and kinetic energy (KE). The structure of a molecule is represented by its PE. The motion of a molecule is defined by its KE. PE function is considered as a quality of the molecule, i.e. fitness function. PE of a molecule can be expressed as an objective function as follows

\[ PE_z = f(x). \]  

where PE\(_z\) means potential energy of molecular structure \(z\) and \(f(x)\) is a fitness function.

The potential energy of a molecular structure is the fitness value of a molecule. Objective function is defined by \(f\) and molecular structure is represented by \(z\). For example, suppose a molecule changes its structure from \(z\) to \(z'\), this is only possible if \(PE_z \geq PE_{z'}\) or \(PE_z + KE_z \geq PE_{z'}\). Kinetic energy (KE) of a molecule defines the degree of local optimum. There are mainly two types of collision among molecules – uni-molecular collision and inter-molecular collision. Uni-molecular reaction can be divided into two types – on-wall ineffective collision and decomposition. Inter-molecular collision can also be categorized into two types – inter-molecular ineffective collision and synthesis.

**On-wall inadequate reaction**

Here, molecules hit the wall and return but there is a change in some of the molecular properties. Figure 1 graphically explains this collision.

Suppose the present molecular structure is \(z\) and the modified molecular structure is \(z'\). Then this change is only possible if

\[ PE_z + KE_z \geq PE_{z'} \]  

since we know that this collision is not much more vigorous. Hence the difference between actual molecule and resultant molecule is small. We get

\[ KE_z = (PE_z + KE_z - PE_{z'}) \times p, \]

where \(p\) lies between [KELossRate, 1], the range of KELossRate is 0 to 1 and \((PE_z + KE_z - PE_{z'}) \times (1 - p)\) is the amount of energy lost when the molecule hits the wall. We keep this energy in buffer, which can be used for decomposition reaction.

**Algorithm 1.** On-wall ineffective (\(K\), buffer)

**Input:** Molecule \(K\) and buffer

1. Find \(\beta_i = \text{Neighbour}(\beta)\)
2. Calculate \(PE_{\beta_1}\)
3. If \((PE_{\beta_1} + KE_{\beta_1} \geq PE_{\beta_i})\)
4. Find \(r\) a random number between [KE LOSS Rate, 1]
5. \(KE_{\beta_1} = (PE_{\beta_1} + KE_{\beta_1} - PE_{\beta_i}) \times r\)
6. Upgrade buffer = buffer + \((PE_{\beta_1} + KE_{\beta_1} - PE_{\beta_i}) \times (1 - r)\)
7. upgrade the molecular structure of \(K\) by \(\beta = \beta_1\)
8. end if
9. output \(K\) and buffer

**Decomposition**

Here, the molecules hit the wall and convert into two or more components. This collision is robust and the structure of resultant molecules is different from the actual molecule. Figure 2 provides a graphical representation of this collision.
The condition for decomposition is \( (NHits[z] - MHits[z]) > \alpha \). In this collision, the shape of the original molecule is \( z \) while those of the resultant molecules are \( z_1' \) and \( z_2' \). Suppose the original molecule has more energy (PE + KE) to capitalize the resulting molecules of PE of \( z \) then the change is allowed as follows

\[
PE_z + KE_z \geq PE_{z_1}' + PE_{z_2}',
\]

Let \( \text{temp}_{1} = PE_z + KE_z - PE_{z_1}' - PE_{z_2}' \).

Therefore, \( KE_{z_1}' = \text{temp}_{1} \times q \) and \( KE_{z_2}' = \text{temp}_{1} \times (1 - q) \), where \( q \) is randomly generated from the interval \([0, 1]\).

Since potential energy of \( z, z_1 \) and \( z_2 \) is approximately the same. Hence, when kinetic energy of molecule \( z \) is very large, then only eq. (3) satisfied. But according to properties of on-wall ineffective reaction, kinetic energy of a molecule always decreases. Hence we have drawn some energy from buffer to favour for satisfying the criteria of decomposition collision. Due to this reason, some energy is drawn from buffer for satisfying the criteria of decomposition.

\[
PE_z + KE_z + \text{buffer} \geq PE_{z_1}' + PE_{z_2}'.
\]

When eq. (4) holds then we can get

\[
\begin{align*}
KE_{z_1}' &= (\text{temp}_{1} + \text{buffer}) \times q_1 \times q_2, \\
KE_{z_2}' &= (\text{temp}_{1} + \text{buffer} - KE_{z_1}') \times q_3 \times q_4,
\end{align*}
\]

where \( q_1, q_2, q_3 \) and \( q_4 \) are randomly generated from the interval \([0, 1]\). Since the buffer already stores a large amount of energy, we multiply two random numbers in both eqs (5) and (6) to ensure that \( KE_{z_1}' \) and \( KE_{z_2}' \) are not too large.

Also, \( \text{buffer} = \text{buffer} + \text{temp}_{1} - KE_{z_1}' - KE_{z_2}' \). If eqs (3) and (4) are not satisfied, the decomposition reaction does not hold and the molecule has its original structure \( z \).

\[\text{Algorithm 2. Decompose (K, buffer)}\]

\textbf{Input:} Molecule \( K \) and central energy

1. Obtain \( \beta_1 \) and \( \beta_2 \) from \( \beta \)
2. Determine \( PE_{\beta_1} \) and \( PE_{\beta_2} \)
3. Let \( \text{temp} = PE_{\beta_1} + KE_{\beta_1} - PE_{\beta_2} - KE_{\beta_2} \)
4. Generate a Boolean variable
5. \( If \ (\text{temp} \geq 0) \) \begin{itemize}
   6. \( \text{Boolean} = \text{TRUE} \)
   7. \( \text{Find a random number p between [0, 1]} \)
   8. \( KE_{\beta_1} = \text{temp} \times p \)
   9. \( KE_{\beta_2} = \text{temp} \times (1 - p) \)
   10. \( \text{Determine new molecule } K_1 \text{ and } K_2 \)
   11. \( \text{Assign } \beta_1, PE_{\beta_1}, \text{and } KE_{\beta_1} \text{to the molecular structure of } K_1 \text{ and } \beta_2, PE_{\beta_2}, \text{and } KE_{\beta_2} \text{to the molecular structure of } K_2 \)
\end{itemize}
6. \( else \ if \ (\text{temp} + \text{buffer} \geq 0) \) \begin{itemize}
   12. \( \text{Boolean} = \text{TRUE} \)
   13. \( \text{Get } k_1, k_2, k_3 \text{ and } k_4 \text{ randomly in meanwhile [0, 1]} \)
   14. \( KE_{\beta_1} = (\text{temp} + \text{buffer}) \times k_1 \times k_2 \)
   15. \( KE_{\beta_2} = (\text{temp} + \text{buffer} - KE_{\beta_1}) \times k_3 \times k_4 \)
   16. \( \text{Upgrade buffer} = \text{temp} + \text{buffer} - KE_{\beta_1} - KE_{\beta_2} \)
   17. \( \text{Assign } \beta_1, PE_{\beta_1}, \text{and } KE_{\beta_1} \text{ to the molecular structure of } K_1 \text{ and } \beta_2, PE_{\beta_2}, \text{and } KE_{\beta_2} \text{ to the molecular structure of } K_2 \)
\end{itemize}
7. \( else \)
8. \( \text{Boolean} = \text{FALSE} \)
9. \( end if \)
10. \( \text{output } K_1, K_2, \text{Buffer and buffer.} \)

\textbf{Inter-molecular ineffective reaction}

Here two or more molecules collide with each other and then return. There is little change in the result. This is similar to the on-wall inadequate reaction. The difference between the two reactions is only the number of molecules. In the first case only one molecule participates in the collision, whereas in the second case two or more molecules participate. In this collision KE is not drawn from the central buffer; there is only interchange among the molecules. Figure 3 is an example of inter-molecular ineffective collision.

Suppose \( z_1 \) and \( z_2 \) are two original molecules and after the inter-molecular ineffective collision we get two different molecular structures \( z_1' \) and \( z_2' \). Since this collision is not vigorous, the molecular structures of \( z_1' \) and \( z_2' \) are not much distinct from the original molecules \( z_1 \) and \( z_2 \). The change is only possible if the following condition is satisfied.

\[
PE_{z_1} + KE_{z_1} + PE_{z_2} + KE_{z_2} \geq PE_{z_1}' + PE_{z_2}'.
\]

Let \( \text{temp}_2 = (PE_{z_1} + KE_{z_1} + PE_{z_2} + KE_{z_2} - PE_{z_1}' - PE_{z_2}') \).

Then, \( KE_{z_1}' = \text{temp}_2 \times r \) and \( KE_{z_2}' = \text{temp}_2 \times (1 - r) \), where \( r \) is a random number between \([0, 1]\).
Algorithm 3. Intermolecular ineffective \((K_1, K_2)\)

**Input**: molecules \(K_1, K_2\) with their profile.

1. Find \(\beta^1 = \text{Neighbour} (\beta^1)\) and \(\beta^2 = \text{Neighbour} (\beta^2)\).
2. Determine \(PE_{\beta^1}\) and \(PE_{\beta^2}\).
3. Let \(temp = (PE_{\beta^1} + PE_{\beta^2} + KE_{\beta^1} + KE_{\beta^2} - (PE_{\beta^1} + PE_{\beta^2}))\).
4. \(\text{If} \ (temp \geq 0)\)
5. Find a random number \(q\) between \([0, 1]\).
6. \(KE_{\beta^1} = temp \times q\).
7. \(KE_{\beta^2} = temp \times (1 - q)\).
8. update the profile of \(K_1\) by \(\beta^1 = \beta^1 + PE_{\beta^1}\) and \(KE_{\beta^1} = KE_{\beta^1}\) and the profile of \(K_2\) by \(\beta^2 = \beta^2 + PE_{\beta^2}\) and \(KE_{\beta^2} = KE_{\beta^2}\).
9. \text{end if}
10. output \(K_1\) and \(K_2\).

**Synthesis**

In this reaction, two or more components collide with each other to form new components. In this collision change is much more effective. Hence the difference between original and resultant molecule is more. Figure 4 is an example of the synthesis reaction. If \(KE_{\beta^1} \leq \lambda\) and \(KE_{\beta^2} \leq \lambda\), this is favourable case for synthesis. In this reaction the original molecules are \(z_1\) and \(z_2\) and the resultant molecule is \(z^*\). This change is applicable if the following condition is satisfied:

\[
PE_{z_1} + KE_{z_1} + PE_{z_2} + KE_{z_2} \geq PE_{z^*}, \quad (8)
\]

Then we get

\[
KE_{z^*} = PE_{z_1} + KE_{z_1} + PE_{z_2} + KE_{z_2} - PE_{z^*}. \quad (9)
\]

If eq. (8) is not satisfied the molecules return to their original structures. In this case, PE does not change, but KE of the resultant molecule is larger than the original molecules. In this reaction, secure molecule has greater ability to escape from local optimum. Any one of the above reactions can hold in each iteration.

Algorithm 4. Synthesis \((K_1, K_2)\)

**Input**: Molecules \(K_1\) and \(K_2\) with their profile.

1. Calculate \(\beta\) from \(\beta^1\) and \(\beta^2\).
2. Determine \(PE_{\beta}\).
3. Generate a Boolean variable.
4. Generate a new molecule \(K\).
5. \(\text{If} \ (PE_{\beta} + PE_{\beta^2} + KE_{\beta} + KE_{\beta^2} \geq PE_{\beta})\)
6. \(\text{Boolean} = \text{TRUE}\)
7. \(KE_{\beta} = PE_{\beta} + PE_{\beta^2} + KE_{\beta} + KE_{\beta^2} - PE_{\beta}\)
8. Assign \(\beta, PE_{\beta}\) and \(KE_{\beta}\) to the profile \(K\)
9. \(\text{Else}\)
10. \(\text{Boolean} = \text{FALSE}\)
11. \text{end if}
12. \text{output} \(K\) and \(\text{Boolean}\).

There are mainly three stages in CRO – initialization, iteration, and final stage. Figure 5 shows a flow chart of the CRO algorithm.

**Proposed method**

In the proposed method, we have used a novel CRO (ICROMSA) to find an approximate solution to the MSA problem. Here the initialization process is improved compared to the basic CRO. ICROMSA generates the new solution using elementary reactions such as on-wall ineffective, synthesis, inter-molecular ineffective and decomposition.

**Initialization scheme**

Here, we have considered the Needleman and Wunsch algorithm to generate the initial population. The process of initialization is shown in Figure 6 and described as follows: (1) Let the problem be defined as given in Figure 7 a. The first pair is selected as given in Figure 7 b. Next, the alignment is computed by considering pairwise alignment approach, as shown in Figure 7 c. In this way, alignment of each pair is estimated. (2) In this step, a random permutation between 1 and \(N\) is generated. Suppose \(N = 4\), then the random permutation is generated between 1 and 4. If the obtained random permutation is \((1, 2, 3, 4)\), then the complete alignment is generated, as shown in Figure 7 d. In the generalized way, \(k\) number of solutions are generated using \(k\) random permutations between 1 and \(N\).

**Molecular representation**

In the MSA problem, dimension of a molecule is equal to the number of profiles \((n)\). Let \(X_i = (X_{i_1}, \ldots, X_{i_d}, \ldots, X_{i_n})\) be the \(i\)th molecule. In profile representation, all the protein elements in MSA are replaced by 0 and the gap is filled by 1. Then the binary sequence in the column is converted into the decimal value which represents the \(X_{i,d}\) for all \(1 \leq d \leq n\).

Figures 8 and 9 show the molecular representation. Initially, the entire protein element in Figure 8 is replaced by 1 and gap is filled by 0 as shown in Figure 9. Then the complete binary sequence in each column is converted into the decimal value which represents the profile as shown in the last row of Figure 9; the resultant decimal sequence of all the profiles is \((5, 0, 0, 0, 10)\).

**Fitness function**

The fitness value of each molecule is determined using sum of pair approach. First, the sum of pair’s symbol is calculated for each column. Then, the sum of all the
Figure 5. Flow chart of chemical reaction optimization.

Start

Input: MSA with $N$ number of sequences. Initialization: structure of molecule = 0

Find alignment between each pair of sequences

Permutation between 1 and $N$

Molecular structure = molecular structure +1

If molecular structure > 200

Stop

Yes

No

Figure 6. Flow chart of an initial generation.

ABCD
BCDE
CDEF
DEFG

(a) (b) (c) (d)

Figure 7. Complete process of an initial generation.

A B C D _
_ B C D E
C D E F _
_ D E F G

Figure 8. Initial solution.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
</tr>
</tbody>
</table>

Figure 9. Encoding scheme.
columns scores is considered to determine fitness function of the corresponding molecule as

\[ T = \sum_{i=1}^{L} T_i, \text{ where } T_j = \sum_{i=1}^{N} \sum_{j=i+1}^{N} \text{Cost}(A_i, A_j), \]

where \( T \) is the total cost of MSA and \( L \) is the number of columns in MSA. \( T_i \) is defined in terms of cost of the \( i \)th column and the number of sequences in complete alignment is \( N \). \( \text{Cost}(A_i, A_j) \) is equal to the alignment score of two sequences \( A_i \) and \( A_j \). \( \text{Cost}(A_i, A_j) \) is defined by PAM matrix. When \( A_i \neq \_ \) and \( A_j \neq \_ \). Also when \( A_i = \_ \) and \( A_j = \_ \) then \( \text{Cost}(A_i, A_j) = 0 \). Finally, \( \text{Cost}(A_i, A_j) = 1 \), when \( A_i = \_ \) and \( A_j \neq \_ \) or \( A_i \neq \_ \) and \( A_j = \_ \) then \( \text{Cost}(A_i, A_j) = 1 \).

### Solution generation

The solution generation process is based on four types of elementary reactions such as on-wall ineffective, synthesis, inter-molecular ineffective and decomposition.

**On-wall ineffective:** Since this reaction is not vigorous, the resultant molecule is similar to the actual molecule. In this reaction, a random position is chosen within a molecule. Thereafter, a random number generated between 0 and \( 2^k \) replaces the corresponding position. For example, a random position is selected, say third. Next, a random number is generated, say 6. Finally, the generated number is replaced with the corresponding position, i.e. 3 replace with 6. Figure 10 shows a graphical representation of on-wall ineffective reaction.

**Decomposition:** This reaction is robust and the difference between resultant and actual is much more. In this reaction, two random positions are selected within the molecule. Thereafter, two right circular shift operations are performed. The first operation is performed using the first randomly selected position and the other by considering the second random position (Figure 11).

**Inter-molecular ineffective:** This reaction is not much effective and the difference between the resultant and actual molecule is less. In this reaction, two random molecules are considered and one random position is selected in each molecule. Then, the values are exchanged between the selected random positions as shown in Figure 12.

**Synthesis:** In this process, two random molecules are considered and one random position is selected within these molecules (say fourth). In the next step, the values are interchanged from the fourth position to the last position between the selected molecules. In the final step, a random molecule is considered for the next generation from the newly generated molecules. Figure 13 shows a graphical representation of the synthesis reaction, while Figure 14 shows a flow chart of ICROMSA. Table 1 lists the implementation parameters of the proposed CRO.

### Dataset

For comparison, we have taken a dataset from BAliBASE version 2.0. BAliBASE 1.0 (ref. 19) contains 142 reference alignments which consist of more than 1000

![Graphical representation of on-wall ineffective reaction.](image10)

![Graphical representation of decomposition reaction.](image11)

![Graphical representation inter-molecular ineffective reaction.](image12)

![Graphical representation synthesis reaction.](image13)

### Table 1. Implementation parameters of ICROMSA

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pop size</td>
<td>200</td>
</tr>
<tr>
<td>Nvars</td>
<td>Number of columns in the sequences</td>
</tr>
<tr>
<td>Initial kinetic energy</td>
<td>40</td>
</tr>
<tr>
<td>Mole coll</td>
<td>0.6</td>
</tr>
<tr>
<td>KE loss rate</td>
<td>0.8</td>
</tr>
<tr>
<td>Buffer</td>
<td>0</td>
</tr>
<tr>
<td>( \alpha )</td>
<td>40</td>
</tr>
<tr>
<td>( \beta )</td>
<td>100</td>
</tr>
</tbody>
</table>
sequences. BAiLiBASE 2.0 (ref. 20) contains 167 reference alignments which consist of 2100 sequences and eight reference sets, which can be described as follows: (i) Reference set 1 contains a small number of intermediate sequences. (ii) Reference set 2 contains totally different or distinct sequences. (iii) Reference set 3 contains a set of divergent sub-families. (iv) Reference set 4 contains extended terminal extension sequences. (v) Reference set 5 contains large interior insertions or deletions. (vi) Reference sets 6–8, contain datasets in which the sequences are repeated. Bali score is used to determine the quality of algorithm. This score defines the level of similarity between manual alignment and resultant alignment. Bali score lies between 0 and 1. If the manual alignment and resultant alignment are the same then the value of Bali score is 1. If both the files are completely dissimilar, then the result is 0. A value between 0 and 1 shows the percentage of similarity match between the manual alignment file and output file obtained from the proposed or existing approach.

**Experimental study**

In this study, simulation is performed using C programming (Linux platform) and graphs are plotted using MATLAB (version 2013). In the performance analysis,
Figure 15 a–d. Performance of CRO and improved CRO per generation w.r.t. reference set 1.

Figure 16 a–d. Performance of CRO and improved CRO per generation w.r.t. reference set 2.
first, the obtained fitness score between the improved CRO and classical CRO is compared. Then, Bali score of improved CRO algorithm is compared with the well-known existing algorithms.

Effect of improved operator in CRO

The CRO algorithm is used in optimization problems for interaction of molecules in a chemical reaction to reach a low-energy stable state. In traditional CRO, initial population is generated randomly, while the improved CRO works with an improved initial population. We have used pairwise alignment algorithm for finding the initial population. To analyse the effect of this proposed initial operator on the algorithms, two types of experiments have been conducted. Both CRO and improved CRO were run. We measured the fitness of each molecule according to fitness function. Figures 15–17 show experimental results with respect to reference sets 1–3.

Comparing the proposed method with GAPAM

To verify the efficiency of the proposed algorithm, we have taken all dataset of GAPAM\(^1\). In GAPAM, the authors keep best Bali score after 20 independent runs. But in this study we have taken average of 10 independent Bali score. We have taken a total of 52 test cases – 18 from reference set 1, 23 from reference set 2, and 11 from reference set 3. These are all taken from Bali base version 2.0. We have taken the approximate results of other methods reported in GAPAM\(^1\). Tables 2–4 provide a summary of the experimental results with respect to reference sets 1–3.

Performance of improved CRO w.r.t. reference set 1:

There are several types of datasets in reference set 1, which differ in length and number of sequences. To show the superiority of the proposed algorithm in terms of Bali score, we have compared it with GAPAM, SB-PIMA, PRRP, HMMT and other well-known techniques. From Table 2, we can see that the proposed algorithm performs better in 13 out of 18 test cases, and GAPAM only five test cases. We take average solution of all methods w.r.t all datasets of reference set 1. The average solution of the proposed method is better than some of the existing methods.

Performance of improved CRO w.r.t. reference set 2:

There are various types of datasets in Bali base reference

---

**Figure 17 a–d.** Performance of CRO and improved CRO per generation w.r.t. reference set 3.
set 2. To show the superiority of proposed algorithm in terms of Bali score, we have compared it with some well-known technique. As seen in Table 3, the proposed algorithm performs better in 19 out of 23 test cases, and GAPAM in only four cases. After experimental analysis, we have seen that the proposed method does not produce best solution in all cases, but in some cases it is close to the best solution. Average score of the proposed method with respect to all test cases is also better than other existing method.

**Performance of improved CRO w.r.t. reference set 3:**
There are many datasets in reference set 3, which is more divergent. Hence residue identities of these datasets are low. Here, we have considered 11 datasets. Table 4 shows that the proposed method is better than other
methods in five test cases: SAGA in one case, RBT-GA in one case, and GAPAM in four test cases. We have calculated average score with respect to all datasets of reference set 3. The average score of the proposed method is better than some of the existing methods taken from GAPAM\(^1\).

**Statistical performance of the proposed method:** We can judge the performance between two different techniques using statistical method. For a comparison between two methods, we have taken Wilcox signed-rank test\(^1\). Table 5 shows statistical results between the proposed method and other methods, where \( W = (W^+ or W^-) \) is the sum of the ranks which is based on the difference between two test variables. We have considered a null hypothesis. Due to property of null hypothesis, when hypothesis is rejected then there is a significant difference between two samples. We have also considered 2.5% level of significance. If the value of \( P \) is less than 0.025, then the null hypothesis is rejected. It means that we can measure the difference between the performances of the executed algorithms, otherwise the difference is not measurable. We have computed Bali score of the proposed method and compared it with existing methods. We find that the proposed method preformed better than MSA-GA, MSA-GA w/prealign and CLUSTAL W for reference set 1. There is also a significant difference for the reference sets 2 and 3. We have found that in a single case there is no notable difference with GAPAM in reference set 1. From this observation, we can conclude that the proposed method is statistically better than other existing algorithms.

**Conclusion**

In the present study, we have proposed an improved CRO algorithm for the MSA problem. The initialization process of CRO has been improved for maintaining the diversity of the solution. In the experimental analysis, benchmark datasets were considered from Bali base 2.0, and the corresponding Bali score was taken which represents the performance of the proposed approach. For the sake of comparison, the proposed approach is compared with several existing approaches such as PRRP, CLUSTAL X, DIALIGN, HMMT, SB-PIMA, SAGA, etc.
RESEARCH ARTICLES

RBT-GA and GAPAM. Simulation results confirm the superiority of the proposed work over others. This implies that the proposed approach can solve the MSA problem in an effective manner.


Received 4 September 2015; revised accepted 17 August 2016

doi: 10.18520/cs/v112/i03/527-538