ISSN No:-2456-2165

DNA Sequence Motif Discovery Using Evolutionary Multi-Objective Differential Evolution Algorithm

Sarker T. Ahmed Rumee
Department of Computer Science and Engineering
University of Dhaka,
Dhaka, Bangladesh

Abstract:- Motif finding in biological sequences is a widely studied yet not fully solved problem. Exact solution to this problem is NP-complete. However, all existing computational methods have been based on some heuristics. On the other hand biological approaches are not efficient both in space and time. Fortunately, large amounts of genome sequencing data are readily available for researchers for analysis and it is possible to build computational model which is also biologically significant. Practical solutions should have highly conserved instances of the candidate motif having biological significance. In this paper, a new approach involving the Multi-objective Genetic Algorithm is used to find motifs in biological sequences. Experiment result shows that the proposed method is successful in discovering planted motif in a group of DNA sequences with high accuracy.

Keywords:- *Genetic Algorithm; Sequence Conservation; Mult-Objective Optimization.*

I. INTRODUCTION

Conserved locations within the regulatory are among the correlated genes are often termed as motifs. Generally motifs are of short (up to 30 nucleotides) length and gapless. Although short in length motifs play major role in gene regulation and identification of these fragments is critical to understand the mechanisms behind DNA transcription process.

To find motifs from sequences several key factors and techniques are taken into consideration. For example, it is observed that genes which are homologous often have similar transcription binding sites and can be found by simple sequence similarity search. However, from computational perspective a potential motif is a set of starting locations within a collection of aligned DNA sequences (we restrict our study to DNA sequences here). So, there can be infinite number of candidate motif locations and searching them computationally are both impractical as it requires exponential runtime.

To solve this problem, many computational methods employing statistical and probabilistic approaches have been developed [7-14] etc. In this paper, we propose a new method to combine the idea of multi-objective optimization with genetic algorithm to find motifs in biological sequences. In the proposed method, an individual is defined by a set of possible starting locations on the multiple sequence alignment of different DNA sequences. The fitness value for an individual is evaluated by optimizing three different objectives – motif length, sequence similarity measure and support (number of sequences where the motif is found). By taking the best possible values in terms of values in three different objectives, our method selects individuals having highly conserved regions. Evaluating our method showed that it is capable of achieving a higher level of prediction accuracy than the competition.

This paper is structured as follows: As you read the first section is the introduction. In Section II, we discuss some necessary background on motif discovery problem and the evolutionary approach. Then section III describes the proposed multi-objective algorithm for motif search. Next section shows the experimental results and compares the results with state of the art algorithm. Finally, we conclude the paper in section VI.

II. BACKGROUND

This section briefly discuss the key terminologies need to understand the proposals made in this paper. It also gives a short overview on the Differential Evolution algorithm and multi-objective optimization, which are the base methods on top of which the proposed system is built.

A. Motif Identification

The motif search problem intends to detect conserved regions from a set of DNA sequences when they are aligned and find good candidate conserved regions or motifs which can eventually indicates transcription factor binding sites.A lot of approaches aims to identify motifs only by considering the promoter region of a set of co-regulated genes from within a single genome. Here the assumption is co-expression of genes are the effects of transcriptional co-regulation.

A more realistic approach would be searching for statically overrepresented motifs in the promoter region of such a set of co-expressed genes. So, searching for motifs are restricted to promoter region of target sequences. Computationally this problem is nothing but a Pattern Search problem.

B. Genetic Algorithm

A genetic algorithms (GA) are inspired by the process of natural selection. GA is commonly used to generate high-quality solutions to optimization and search problems by using operators closely mimicking the evolutionary process such as mutation, crossover and selection [21].

An evolutionary algorithm starts with a set of candidate solutions (termed as individuals) which are often randomly initialized. This initial set of candidate solutions is called the Population. Each candidate solution has a set of parameters (variables) known as Genes. Often these genes are represented using the string representation. Generally presence or absence of certain parameters are denoted by these genes with binary strings of 0s and 1s, mimicking the gene encoding the chromosome.

The fitness function determines an individual's value in regards to the solution goal of the problem in hand. The solution goal may be is to maximize or minimize the fitness score of an individual. As a whole it represents whether an individual is fit to be considered in the next generation of optimization or not. The probability of an individual to be used for reproduction is based on its fitness score. So the selection phase aims to find the fittest individuals and pass then to the next generation.

Crossover is used to introduce the idea of mating or individuals and creation of offspring in real life. At the implementation phase of GA, a random point within the candidate within the genes. Offspring (new candidate individuals) are created by exchanging the genes of parents among themselves until the crossover point is reached.

Similarly, *Mutation* is also an operator to create random variance. Here instead of using multiple individuals, a single candidate solution is randomly mutated to mimic the biological mutations occurring during the evolutionary process. Generally mutations happened with a low random probability. In true implementation sense, this implies that some of the bits in the bit string representing an individual is randomly flipped.

At the end, a GA terminates if the set of candidate solutions (population) converges to an optimal set (does not produce offspring which are fitter from the previous generations).

C. Multi-Objective Optimization

Multi-objective optimization (MOO) approaches involve more than one objective function to be optimized simultaneously. Typically, it is not possible to get a solution which optimizes all objectives to their optimum values. However, the trade-off here is we want to find solutions which are good overall in all objectives, where a single objective value may not be the optimal. So, there can exist a (possibly infinite) set of optimal solutions forming what is called the Pareto Optimal set.

We call a solution non-dominated or Pareto optimal if none of the objective functions can be improved in value without degrading one or more of the other objective values.

III. METHOD

This paper proposes a multi-objective version of the well-known Differential Evolution genetic algorithm to find a large number of motifs from DNA sequences. The proposed method work on three objectives, which will be discussed in the objectives subsection.

In this section we describe our proposed method to solve motif finding problem. At first, we define structure of individuals (Population members), selection scheme, genetic operators (crossover and mutation). Then we define the objectives that we want to optimize and details of the proposed approach.

A. Structure of Individuals

An individual represents the starting locations of potential motif on the all target sequences. Each individual is divided into n genes where there are n sequences in our data set. The individual also contain the length of this potential motif. So the size of an individual is ultimately (n+1). Apart from the first location which represents the length, genes are positional. The first gene deal with the first sequence. The second one deals with the second sequence, and so on. The structure of the individual is depicted in the figure below.

length	S_1	S_2	. . . . . .	S_n

Fig 1:- Representation of an Individual

Here S_i denotes the location of the motif instance in the i^{th} sequence and Length denotes motif length corresponding to that individual. The value of this field can change from 5 to 40 as we restricted algorithm to find a motif with length in between them. The Length field gives greater flexibility compared to the other evolutionary approaches. Here, each individual has fixed length, however with the presence of variation operators (mutation and crossover) individual phenotype (Motif) has a variable length.

B. Initial Population Generation

We start with a fixed number of randomly initialized individuals, which forms the preliminary population. Then these individuals are evolved over a series of generations.
Each member of the initial population is also a potential motif having the structure as described in the previous section. We start with a population of 200 individuals as we found out experimentally that this is an ideal population size for the problem in hand.

C. Objectives and Fitness

Fitness of an individual is assessed based on its value in three objectives: motif length, support and similarity. We restricted our formulation to three objectives as increasing the number of objectives from three increases the complexity of the problem to a great extent. As in that case most of the individuals will non-dominated by each other and try to access the pool of non-dominated solutions, called Archive. Now we discuss how the three objectives we want to optimize play the role in defining fitness of an individual.

- *Motif Length*

It is desirable that we find motifs which are long, because a motif with longer length are more likely to be biologically significant and has a less chance to occur randomly by chance.

- *Similarity*

Similarity measures the degree conservation of the candidate motif (individual in the population) in a multiple sequence alignment. To calculate the score of a candidate motif at first Position Weight Matrix (PWM) is generated.

PWM has four columns representing four possible nucleotides (A, T, G or C) of a DNA sequence. Number of rows will be the same as the length of the candidate. For example, for the four DNA sequences shown below the position weight matrix is mentioned in Table. 1.

GACTTCGTC
GTGTACGAC
GCGTGCATC
GAGTCTACT

Table 1:- Example Candidate Motif in a Set of DNA Sequence

The corresponding PWM is:

	1	2	3	4	5	6	7	8	9
A	0	0.50	0	0	0.25	0	0.5	0.25	0
T	0	0.25	0	1.0	0.25	0.25	0	0.50	0.25
G	1.0	0	0.75	0	0.25	0	0.5	0	0.25
C	0	0.25	0.25	0	0.25	0.75	0	0.25	0.50

Table 2:- Position Weight Matrix of Table 1

Next, for each column of the PWM, the nucleotide with maximum weight is determined. This value is termed as the maximum value (*max*). The objective value score is determined by the following formula:

***Similarity* = $\sum max_i /N$,**
Where *N = no of columns in the PWM*

- *Support*

We assume that a motif may not be present in all the sequences. To handle this scenario, we consider the value of *support* as the third objective in the proposed method. Number of sequences containing a motif is termed as the support of that motif sequence.

D. Configuration of Genetic Operators

The proposed method uses very simple crossover and mutation parameters. Crossover rate was fixed at 0.9. It actually dominates how often the recombination of the parent individuals take place to form the child population. This high cross over rate ensures that the set of individuals are not biased or skewed towards certain individuals from the population.

Each individual is defined as a list of starting positions of a candidate motif. As a result, to implement mutation simple left or right shift operations were used thereby create new candidate motif positions.

E. Algorithm Details

Here, based on the parameter choice discussed so far, our work towards using the Differential Evolution genetic algorithm and converting it into a multi-objective algorithm is discussed formally. At first we introduce few terminologies used to depict the algorithm (Table 3) and then present the

Generation Number	G
Problem Dimension	D
No of initial population	NP
Maximum generation count	MAX_GEN
A single individual	$X_{i,G} = X^{1}_{i,G},\ldots\ldots\ldots\ldots,X^{D}_{i,G}$
Initial population	$P_G = X_{1,G},X_{2,G},\ldots\ldots,X_{NP,G}$
Crossover Ratio	CR
Rate of Mutation	F
External archive of G	A_G

Table 3:- Key Terms and Notations Used in Algorithm 1

```
G <-0
PG <- Randomly Initialize
AG <- PG
While G < MAX_GEN do
    For all i where 1<=i<=NP do
        jrand <- (int)random num in [0,D)
        for all j where 0<=j<=D-1 do
            /** Cross Over **/
            urand <- random numb in [0,1)
            if urand < CR or j = jrand then
                /** Mutation **/
                Vi,j = Xbest,j+F*(Xr1,j-Xr2,j)
                    +F(Xr3,j-Xr4,j)
            else
                Vi,j = Xi,j
            endif
        endfor
        Copy better of Trial(Vi,G)and Target(Xi,G) [from
        SELECTION phase] to generation G+1
    endfor
endwhile

SELECTION Phase
for all i where 1<=i<=NP do
    if Xi,G dominates Vi,G then
        reject Vi,G
    else if Vi,G dominates Xi,G then
        Xi,G+1 = Vi,G, update AG
    else
        Xi,G+1=less-crowded(Xi,G+1,Vi,G),ref: AG
    endif
endfor
```

Algorithm1:- Multi-Objective DE to Find Motif

IV. RESULTS

This section discusses in details the evaluation of the proposed genetic algorithm for motif finding in biological sequences. At first the experiment environment and the required parameter setting for the initiation and running of the genetic algorithm are discussed. Then a brief overview of the data sets is given. Finally, the outcome of the experiment is presented and discussed.

A. Environment Setup

All the experiment is conducted on a PC having 2.6 GHz Intel Core i5 processor, 16GB of main memory and running the Ubuntu 16.04 as the operating system.

B. Parameter Setup

Multi-objective Differential Evolution is a stochastic search method. So, results obtained using such method highly depends on the optimal parameter setting.

Two other parameters which are very much important are - *Threshold* and *AT_{count}*. For a particular individual *Threshold* is defined to be (*Length * 0.5*). Then we only consider those sequences having distance to consensus sequence smaller or equal to this value.

Paramters	*Value*
Population Size	300
Archive Size	600
Maximum Generations	5000
Crossover Probability (CR)	0.9
Mutation Probability (F)	0.3
Threshold_Support	*floor*(No of Sequence *0.5 + 0.5)
AT_{Count}	0.6

Table 4:- Parameter Setting

As we described in the previous chapter this defines individual performance in the second objective function support. The parameter *AT_{count}* defines the percentage of A and T in a candidate motif defined by an individual. If the percentage crosses 60%, then we consider that individual as the *TATA box*.

Another related parameter is *Threshold_support*. It is related to the objective *support* and if the value of *support* decreases from the *Threshold_support*, then we do not consider that individual for our operation. The parameter settings are summarized in Table 4.

C. Data Sets

For experiment, three standard data sets were used, which were also used as benchmark in analyzing performance of state of the art computational approaches for sequence motif identification approaches.

First two data sets were taken from the TRANSFAC [3] database, named *yst04r* and *yst08r*, representing sequence data of yeast species.

Apart from these data sets, sequence data taken directly from the yeast transcriptional regulation site is also used. This data set was taken from SCPD (Promoter Database of Saccharomyces cerevisiae) [4].

D. Result of Experiment on Yst04r

The data set yst04r contains 7 sequences. Each sequence is 1000 bases long. A subset of the non-dominated solutions found by our method is shown in the following table (Table. 5).

For the sake of comparison we include two columns for the *Similarity* value, one for our scheme and the other for the method proposed by Mehmet [5]).

Support	*Length*	*Similarity*	
		Mehmet's Scheme	*Proposed Method*
4	24	0.76	0.81
4	20	0.78	0.80
4	15	0.81	0.85
5	15	0.82	0.81
5	14	0.84	0.84
6	14	0.77	0.78
6	13	0.81	0.84
7	9	0.80	0.79
7	8	0.84	0.82

Table 5:- Experiment Results on Data Set yst04r

Although the solutions vary in each of the objectives it was done so to show a direct comparison between these two methods.

It is also justified to compare two methods based on *similarity* value while keeping other objective values constant, as we know the closer the *similarity* value is to 1. 0, it is more probable for the corresponding individual to be discovered as a motif.

Apart from these, some of the promising results explicitly found by the proposed method are listed in Table 6.

Support	*Length*	*Similarity*
4	16	0.82
5	20	0.80
6	12	0.83
6	19	0.76
7	14	0.81

Table 6:- Novel Finding of Proposed Method on Yst04r Data

E. Result of Experiment on Yst08r

The second data has 11 sequences, each of which is 1000 base pairs in length. The comparison results of the experiment outcome for the proposed method and the Mehmet's scheme [5] are listed in Table. 7 below.

Support	*Length*	*Similarity*	
		Mehmet's Scheme	*Proposed Method*
7	20	0.75	0.81
7	15	0.84	0.85
8	15	0.79	0.84
8	14	0.83	0.84
8	13	0.85	0.88
9	13	0.82	0.85
9	12	0.84	0.84
10	12	0.79	0.84
10	11	0.82	0.86
11	11	0.80	0.85

Table 7:- Experiment Results on Data Set Yst08r

Like the ys04r data set, for yst08r data set the proposed method explicitly found motifs of certain lengths and similarity score as listed in Table 8.

Support	*Length*	*Similarity*
7	16	0.84
7	14	0.83
8	20	0.81
8	17	0.82
10	16	0.78
10	15	0.80

Table 8:- Novel Finding of Proposed Method on Yst08r Data

F. Result From Yeast Transcriptional Regulation Site

The database SCPD (Promoter Database of Saccharomyces cerevisiae) [4] contains the promoter regions of 6000 genes and ORF in yeast genome. The regulon in the database is a set of co-regulated genes whose promoter share binding sites for the same transcription factors.

We extracted the promoter regions from the two regulon families from SCPD. The families chosen are MCB and LEU3. The reason behind choosing these two families was to show comparison with the result found by Paul and Iba [6].

For MCB transcription factors. We extracted six sequences from the positions -500 to +50 of transcription start site of regulated genes of Saccharomyces cerevisiae.

The motifs embedded in these sequences are ACGCGT. ACGCGA.CCGCGT. TCGCGA.ACGCGT, ACGCGT and the consensus sequence is WCGCGW. Here we found the following motifs: ACGCGT. ACGCGT ACGCGT. ACTCGA. ACGCGT. ACGCGT and the consensus sequence is ACCCCT. The following table (Table. 9) shows the results found by our method and Paul and Iba's [6] method.

Proposed Method	*Paul and Iba's Method*
ACGCGT	ACGCGT
ACGCGT	ACGCGT
ACTCGA	ACTCGA
ACGCGT	ACGCGT

Table 9:- MCB Transcription Factor Binding Sites Detection Results

On the other hand for the *LEU3* transcription factor, we extend two sequences from position -500 to +500 of the transcription start site of the two regulated genes of Saccharomyces cerevisiae. The consensus motif is CCGNNNNCGG. The motifs found by the proposed method and the Paul and Iba's method are shown in Table. 10.

Proposed Method	*Paul and Iba's Method*
CCGGGACCGG GCGGAACCGG	CCGGAACCGG CCGGGACCGG
CCGTAACCGG CCGGAACCGG	CCGGAACCGG CCGTAACCGG

Table 10:- LEU3 Transcription Factor Binding Sites Detection Results

It is evident from the tabular data that both the proposed method and Paul and Iba's method found almost the same result.

V. RELATED WORK

Finding motifs in biological sequences has been studied heavily in the last decade or so. Here we limit our discussions only to the closely related approaches.

Among the deterministic approaches graph and tree based methods were most successful [7, 8, 9, 10]. However these methods miss many biological factors associated with the presence of motif and hence produce a lot of false positives and false negatives.

Probabilistic approaches [11, 12, 13, 14] produce more false positives, however gives less false negatives. As a result these approaches laid the foundation for more sophisticated evolutionary algorithms. EM algorithm [14] developed by Lawrence and Reilly is a greedy algorithm based method to find motifs. The EM algorithm finds motifs in unaligned biological sequences. They collects candidate motifs for a given weight matrix using random walk (expectation step) and updates the candidate list with motifs of higher expected values (maximization step). Similarly, GibbsDNA [13] tries to maximize the similarity among subsequences to identify potential motif locations.

Evolutionary approaches uses the natural selection mechanism of evolutionary theory to drive its operation. Genetic algorithms [15, 16, 17, 18, 19, 20] are found to be finding motifs in sequences with high accuracies. In this work, we applied the genetic algorithm based approach with the principle of multi-objective optimization and found better results compared to state of the art genetic algorithm for motif finding algorithm MOGAMOD [5].

VI. CONCLUSION

In this paper, we proposed a new technique based on Multi-objective genetic algorithm to look for motifs in DNA sequences. Evaluation results proved the effectiveness of the proposed method.

However, there are few possibilities to further extend this work which we plan to investigate next:

- At present our method cannot work with the gapped motif which we are currently working on.
- The similarity measure can be improved to improve the proposed method to make it work with all kinds of sequences.
- Automatic evolution of control parameters of the Differential Evolution parameters can be an important improvement of the proposed method as its success depends on correct choice of parameter values.

REFERENCES

[1]. G. Eason, B. Noble, and I.N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529-551, April 1955. (*references*)

[2]. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[3]. TRANSFAC, http://genexplain.com/transfac/.

[4]. SCPD--Saccharomyces cerevisiae promoter database, https://www.hsls.pitt.edu/obrc/index.php?page=URL1101827596

[5]. Kaya, Mehmet. "MOGAMOD: Multi-objective genetic algorithm for motif discovery." Expert Systems with Applications 36.2 (2009): 1039-1047.

[6]. Paul, Topon Kumar, and Hitoshi Iba. "Identification of weak motifs in multiple biological sequences using genetic algorithm." In Proceedings of the 8th annual conference on Genetic and evolutionary computation, pp. 271-278. ACM, 2006.

[7]. Eskin, Eleazar, and Pavel A. Pevzner. "Finding composite regulatory patterns in DNA sequences." Bioinformatics 18, no. suppl_1 (2002): S354-S363.

[8]. Liang, Shoudan, Manoj Pratim Samanta, and B. A. Biegel. "cWINNOWER algorithm for finding fuzzy DNA motifs." Journal of bioinformatics and computational biology 2, no. 01 (2004): 47-60.

[9]. Marsan, Laurent, and Marie-France Sagot. "Algorithms for extracting structured motifs using a suffix tree with an application to promoter and regulatory site consensus identification." Journal of computational biology 7, no. 3-4 (2000): 345-362.

[10]. Vanet, Anne, Laurent Marsan, Agnes Labigne, and Marie-France Sagot. "Inferring regulatory elements from a whole genome. An analysis of Helicobacter pyloriσ80 family of promoter signals." Journal of molecular biology 297, no. 2 (2000): 335-353.

[11]. Down, Thomas A., and Tim JP Hubbard. "NestedMICA: sensitive inference of over-represented motifs in nucleic acid sequence." Nucleic acids research 33, no. 5 (2005): 1445-1453.

[12]. Hertz, Gerald Z., and Gary D. Stormo. "Identifying DNA and protein patterns with statistically significant alignments of multiple sequences." Bioinformatics (Oxford, England) 15, no. 7 (1999): 563-577.

[13]. Lawrence, Charles E., Stephen F. Altschul, Mark S. Boguski, Jun S. Liu, Andrew F. Neuwald, and John C. Wootton. "Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment." science 262, no. 5131 (1993): 208-214.

[14]. Lawrence, Charles E., and Andrew A. Reilly. "An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences." Proteins: Structure, Function, and Bioinformatics 7, no. 1 (1990): 41-51.

[15]. Che, Dongsheng, Yinglei Song, and Khaled Rasheed. "MDGA: motif discovery using a genetic algorithm." In Proceedings of the 7th annual conference on Genetic and evolutionary computation, pp. 447-452. ACM, 2005.

[16]. Congdon, Clare Bates, Charles W. Fizer, Noah W. Smith, H. Rex Gaskins, Joseph Aman, Gerardo M. Nava, and Carolyn Mattingly. "Preliminary results for GAMI: A genetic algorithms approach to motif inference." In 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, pp. 1-8. IEEE, 2005.

[17]. Fogel, Gary B., Dana G. Weekes, Gabor Varga, Ernst R. Dow, Harry B. Harlow, Jude E. Onyia, and Chen Su. "Discovery of sequence motifs related to coexpression of genes using evolutionary computation." Nucleic Acids Research 32, no. 13 (2004): 3826-3835.

[18]. Liu, Falcon FM, Jeffrey JP Tsai, Rong-Ming Chen, S. N. Chen, and S. H. Shih. "FMGA: finding motifs by genetic algorithm." In Proceedings. Fourth IEEE Symposium on Bioinformatics and Bioengineering, pp. 459-466. IEEE, 2004.

[19]. Garbelini, Jader M. Caldonazzo, André Y. Kashiwabara, and Danilo S. Sanches. "Sequence motif finder using memetic algorithm." BMC bioinformatics 19, no. 1 (2018): 4.

[20]. Mohanty, Satarupa, Rohan Chauhan, and Biswajit Sahoo. "Genetic Algorithm for Planted Motif Search (I, d) with Iteration on Average Value." International Journal of Soft Computing 13, no. 1 (2018): 18-24.

[21]. Genetic Algorithms, https://en.wikipedia.org/wiki/Genetic_algorithm.