Motif Finding with Application to the Transcription Factor Binding Sites Problem

Vilas Machhi
Student, BVM Engineering College

Maulika S Patel
Associate Professor & Head, Department of Computer Engineering, G H Patel College of Engineering & Technology

Jyoti Degama
Alumnus, Department of Computer Engineering, G H Patel College of Engineering & Technology

ABSTRACT
DNA sequencing of different species has resulted in the generation of huge amount of biological data. There is an increasing need to develop computational techniques to search for relevant information in the DNA data. Discovering motifs involves determining short sequence segments which have a high probability of repeated occurrences over many sequences in different species. Motifs are useful in finding transcription factor binding sites, transcriptional regulatory elements and so on. Transcription factor binding sites (TFBSs) is important for understanding the genetic regulatory system. Our method is based on the Ant Colony Optimization (ACO) and Gibbs sampling algorithm to discover DNA motifs (collections of TFBSs) in a set of DNA-sequences. We first applied an ACO algorithm to find a set of better candidate positions for the motif. The resultant positions are given as input to the Gibbs sampler method for calculating score for each sequence. Based on the score, motif for TF binding sites is identified.

General Terms
Bioinformatics, Pattern Recognition

Keywords
Motif, transcription factor binding sites (TFBSs), Gibbs Sampling, Ant colony Optimization**.**

1. INTRODUCTION
A DNA motif is a nucleic acid sequence pattern having biological significance such as being DNA binding sites for a regulatory protein, ribosome binding, mRNA processing (splicing, editing, polyadenylation) and transcription termination. Pattern is short (5 to 20 base-pairs (bp) long) and is known several times within a gene [1]. Characteristics of motifs are:

- They comprise of patterns of length 10 to 25 bases, with repeated occurrences
- They are statistically over-represented in regulatory regions
- They are small, have constant size, and are repeated very often.

Motifs are patterns found in biological sequences that are essential for understanding the function of genes, human diseases, drug design, etc. They are important for identifying transcriptional regulatory elements, transcription factor binding sites, etc. making the problem of identifying motifs an important one in biology [2]. Suppose a transcription factor (TF) controls five different genes. Each of these genes should have binding sites for TF in their promoter region. Now suppose we are given the promoter regions of the five genes g1, g2, g3, g4, g5. We cannot find the binding sites of TF, without knowing about them a priori. Binding sites are similar to each other, but not necessarily identical. This is the motif finding problem.

Gene expression is regulated by Transcription Factors [15], to their corresponding binding sites. Transcription Factors bind to specific DNA sequences, namely Transcription Factor Binding Sites (TFBSs) [4, 5], to initialize, assist, or suppress transcriptional activity. TFBSs are usually small DNA sequences in the range of 6 to 30 bps. As of now, the most accurate and reliable method for detecting TFBSs remains biological experiments such as DNAse footprinting assay [6] and Electrophoretic Mobility Shift Assay (EMSA) [7]. These methods are labor intensive and time consuming.

Mohan Das and Dai surveyed various approaches including the popular Gibbs sampling method and machine learning techniques for motif finding [16]. In [3], a hybrid method that used ACO and Gibbs Sampler method for finding motif in protein sequences. In this paper, we propose such a hybrid method which uses ACO [8] and Gibbs Sampler method [9, 16] to find a motif in DNA sequences. By using ACO algorithm, we can find better starting positions of sequences for motif finding. These positions of sequences provide a set of candidates for the Gibbs sampler method to get a better solution for motif identification.

2. THE GIBBS SAMPLER METHOD FOR MOTIF FINDING
Gibbs sampling [10] is a statistical technique related to Monte Carlo Markov Chain sampling. Gibbs sampling randomly choose a beginning position in each sequence and built position weight matrix for that sequence. From the position weight matrix, the score for each sequence is calculated. Randomly one of the sequences is selected and removed based on which the matrix is updated. It is seen that the scores at most positions are not good enough and also requires more time for processing.

The Gibbs sampling algorithm uses two data structures, one for the probabilistic model and another for the set of positions. The probabilistic model consists of two components. One component is applied to calculate the variables of nucleotide frequencies at each position i of a set of subsequences, $1 \leq i \leq w$, where w is the length of motif. Second component describes all other background positions. The set of positions Pk, $1 \leq k \leq n$, constitutes the alignment, where Pk denotes the starting position of pattern xk in sequence k. The concept of this algorithm is as follows [9, 14]:

Input: A set S of n sequences S1, S2,… , Sn, and motif length w.

Output: A motif x1, x2, …, xn, each of length w, where each xi is a substring of Si.

[1] Randomly choose a beginning position pi in each sequence Si, $1 \leq i \leq N$. Let xi be a substring of Si with length w and starting from position pi.

[2] Randomly select one sequence, S*, in S , whose substring is xs*. Let U ={x1, x2,…, xn} − {xs*}.

[3] With U, create a 4 × w probability matrix M. In the matrix, each row represents nucleotides. Note the number of nucleotides is 4. Mr,j represents an occurrence frequency, which is the number of sequences that nucleotides r appears at position j.

[4] Let Rl be a substring of S* with length w and starting at position l. For each Rl, calculate a score ql, which involves the frequency matrix M and the background distribution B as in equation (1).

$$ql = \sum_{r=1}^{4} \sum_{j=1}^{w} Mr, j \log \frac{Mr, j}{Bj} \qquad (1)$$

[5] Randomly choose a starting position l in S* with probability proportional to ql. The new starting position l will construct Rl which is a new substring of S* with length w. Then xs* is replaced by Rl.

[6] If the best solution has not been changed after some predefined iterations, terminate the algorithm; otherwise, go to step 2.

3. ANT COLONY OPTIMIZATION ALGORITHM (ACO)

Swarm intelligence is an approach inspired from the collective behavior of insects or other animals for problem solving. Few of the algorithms in this category are the artificial bee colony algorithm, Cuckoo search algorithm, termite hill algorithm and the ant Colony Optimization algorithm (ACO) [8] are proposed and applied to a wide range of problems.

The ACO algorithm is based on the behavior of real ants which somehow find the shortest path to the food. In an experiment carried out by Goss et. Al, it became evident that ants could find the shortest path to the food source. This is the result of the interaction amongst the ants by understanding the intensity of the chemical component, pheromone, dropped by each ant on the path. More the ants moving on a particular path, more favorable the path is. The ACO algorithm simulates the natural behavior of ants, including mechanisms of cooperation and adaptation.

ACO algorithms are based on the following concepts [11]:

- Every path followed by an ant is associated with a feasible solution for a given problem.
- Amount of pheromone deposited on path is proportional to the quality of the corresponding feasible solution for the target problem.
- When an ant has to decide between two or more paths, it uses the pheromone deposition intensity to select the path. (figure 1).

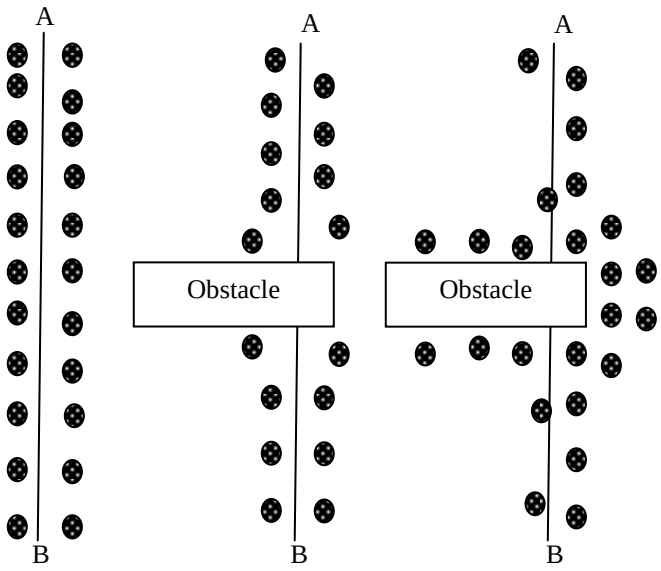

Figure 1: (a) Real ants travel a path between A and B. (b) Block hampers the movements of the ants on the path. Each of the ants' select one way, left or right, with equal probability. (c) More ants traveling on the shorter path, pheromones are deposited more quickly on the shorter path.

ACO meta-heuristic is applied to a wide range of problems such as the Travelling salesman problem, the vehicle routing problem, quadratic assignment problems, scheduling problems, and more. ACO have been successful in providing optimal solutions to many NP-Hard problems. However, application of ACO in the bioinformatics domain is limited and yet to be explored.

4. OUR METHOD

There are two steps in the Gibbs sampler method. First, Gibbs sampler method randomly chooses a beginning position Pi of each sequence Si in S. By using these random beginning positions, it makes slow progress. Second, Gibbs sampler method calculates the score ql at each position l in sequence. Scores at most positions are not good enough to be paid more attention. Most positions do not provide any help to find a better solution.

We apply ACO algorithm to find a set of better initial positions in each sequence. Then we apply the Gibbs sampler method with the set of better initial positions as the inputs. And we calculate the scores of those initial positions in sequences instead of all positions [figure 2].

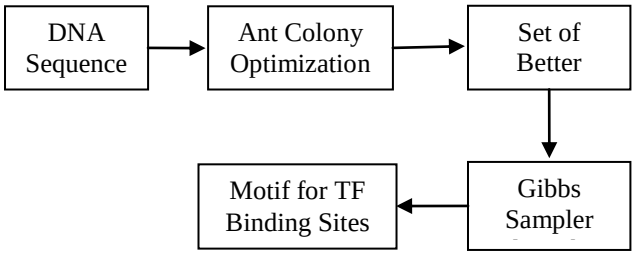

Figure 2: Flow diagram of the proposed algorithm. DNA sequence is given to ACO algorithm to explore better initial positions to be used by the Gibbs sampler to find the motif for TF binding sites.

By using the ACO algorithm, efficiency and quality of Gibbs sampler method is improved. The total required computing time is reduced, because the beginning positions are randomly chosen in the Gibbs sampler method, which takes very long time to converge to a better solution.

We applied ACO algorithm to find motif in input sequences. Ant chooses path depends on pheromone probability. The probability of pheromone is given by follows equation:

$$Qk(lu) = \frac{[\tau_{lu}(t)]^{\alpha}}{\sum_{u \in C} [\tau_{lu}(t)]^{\alpha}} \quad (2)$$

$Q_k(lu)$ represents the probability that ant k chooses the character u at position l. C is the character set of input sequences. α is a parameter which is used to determine the influence of the pheromone trails. $\tau_{lu}(t)$ is explained as follows.

$$\tau_{lu}(t+1) = (1-\rho).\tau_{lu}(t) + \sum_{k=1}^{m_{lu}} \Delta\tau_{lu}^{k}(t) \quad (3)$$

The left side $\tau_{lu}(t+1)$ means the consistency of pheromone on the character u at position l. m_{lu} denotes the total number of ants which carry the character u at position l. The parameter α, $0 < \alpha < 1$, is the rate of the pheromone trails evaporation. $\Delta\tau_{lu}^{k}(t)$ denotes the variable of pheromone on the character u at position l that ant k has chosen.

The travel of ants between nodes depends on the pheromone. We assume that each ant travels from the starting point to the terminal point and passes through w middle nodes. On each node, there are |C| kinds of foods, where |C| denotes the number of characters in set C. Each ant travels from the starting point A. When each ant passes through middle nodes, it chooses to pick up one of the |C| foods depending on the consistency of pheromone. After each ant arrives at the terminal point B, it carries w foods which are collected from middle nodes. Those w foods are constructed into a string to form a *sample motif*.

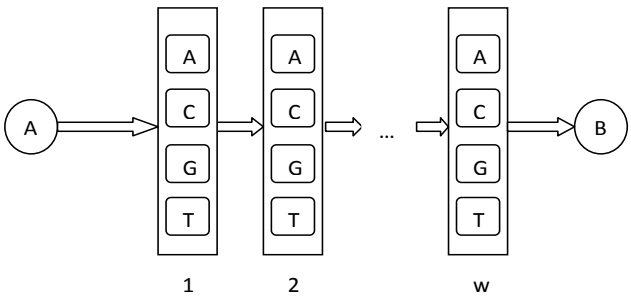

Figure 3: Each ant k travels from the starting point *A* to the terminal point *B*. When each ant passes through middle nodes, it picks up one of the four foods A, C, G, and T.

The concept of this Algorithm is as follows [3]:
Input: A set S of n sequences S1, … , Sn, and motif length w.
Output: x1, … , xn, each of length w, where xi is a substring Si.
[1] Set parameters and initialize pheromone trails.

[2] Each ant k randomly constructs a sample x^k with length w. The probability of choosing the character is calculated by Formula 2.

[3] Each ant k compares sample x^k with each sequence Si to find the best matched substring x_i^k in Si. Then each ant k gets a set $V^k = \{x_1^k, x_2^k, ..., x_n^k\}$.

[4] Apply the score function, Formula 1, to calculate the score q^k of each substring set V^k.

[5] Update the pheromone with Formula 3.

[6] Find the best score q^b among all q^k s. And update the best sample x^b accordingly. If the best sample x^b is not changed for some predefined iterations, go to Step 7; otherwise, go to Step 2.

[7] Compare x^b with each sequences Si to find a set Ai. And let xi be the best position in Ai.

[8] Randomly select one sequence, S*, in S, whose best position is xs*. Let U = {x1, x2, · · · , xn} − { xs* }.

[9] With U, create a |C|×w probability matrix M, where each row represents a character in the set C. Mr,j represents an occurrence frequency, which is the number of sequences that nucleotides r appears at position j.

[10] For each candidate position Rl of S*, calculate the score of ql by using formula 1.

[11] Randomly choose a starting position l in S* with probability proportional to ql. Update the best substring set V^b if the best score q^b is changed, where V^b denotes the substring set of the best motif.

[12] If the best substring set V^b is not changed after some predefined iterations, terminate our algorithm; otherwise, go to Step 8.

5. RESULTS

For testing the hybrid method, the H3N2 virus data set is chosen [17] which consist of 685bytes. We apply ACO algorithm to find a set of better initial positions for the Gibbs sampler method. The uncertainty arising while randomly choosing a set of beginning positions is reduced which results in convergence to a better solution. Table-1 shows scores of each motif at a particular position in the sequence. It is interesting to observe that the motif "CAGAAC" at position 2 in the H3N2 DNA sequence has a score of 8.55 which is the highest. It is biologically understood that CAGAAC has important functionality.

Table 1. The first column is obtained after applying ACO to DNA sequences. The second and third columns are obtained by applying Gibbs sampler on the sequences in column 1.

Sequence	Position	Score
TCAGAA	1	1.2476851851851853
CAGAAC	**2**	**8.555555555555555**
AGAACC	3	1.8148148148148149
GAACCA	4	4.76388888888889
AACCAG	5	5.703703703703703
ACCAGT	6	0.48611111111111116
CCAGTT	7	0.9074074074074074
CAGTTA	8	7.0
AGTTAT	9	5.703703703703703
GTTATA	10	2.3333333333333335
TTATAA	11	3.2083333333333335
TATAAA	12	5.703703703703703
ATAAAT	13	2.138888888888889
TAAATT	14	2.3333333333333335

AAATTT	15	3.5
AATTTA	16	4.666666666666667
ATTTAT	17	4.277777777777778
TTTATC	18	1.5555555555555556
TTATCA	19	2.041666666666667
TATCAT	20	6.654320987654321
ATCATT	21	1.1666666666666667
TCATTT	22	0.875
CATTTC	23	6.222222222222222
ATTTCC	24	2.7222222222222228
TTTCCT	25	2.117283950617284
TTCCTT	26	1.3611111111111114
TCCTTC	27	0.875
CCTTCT	28	1.8148148148148149
CTTCTC	29	2.419753086419753
TTCTCC	30	2.041666666666667
TCTCCA	31	1.058641975308642
CTCCAC	32	3.3271604938271606
TCCACT	33	0.6805555555555557
CCACTC	34	0.9074074074074074
CACTCC	35	5.4444444444444455
ACTCCT	36	1.058641975308642

6. CONCLUSIONS

A single nucleotide difference can alter protein function in such a way that it fails to function normally. Single nucleotide changes have been linked to hereditary differences in height, facial structure, pigmentation, brain development, and many other striking morphological differences; due to single nucleotide changes, hands can develop structures that look like toes instead of fingers. Therefore motif finding is important in biology.

First, we apply ant colony optimization algorithm which give set of better initial position for Gibbs sampling. Gibbs sampling takes that initial position and calculate score for each sequence. In our algorithm Gibbs sampling doesn't take randomly selected sequence for calculation so, time required for finding motifs using our algorithm is reduced drastically. Given the rate at which the DNA and Protein databases are growing, it is important to develop efficient algorithms and computational tools for finding motifs. Finding motifs remain a problem of interest to computational biologist and other researchers given its importance in drug design and early detection of diseases. In this paper, the method can be tested and applied to various data sets and the efficiency in terms of time could be measured. Future work involves finding the efficiency of proposed algorithm on various data sets.

7. REFERENCES

[1] Rahul Chauhan, Dr.Pankaj Agarwal,A Review: Applying Genetic Algorithms for Motif Discovery, (2012).

[2] HieuDinh, Sanguthevar Rajasekaranand Vamsi K Kundeti, PMS5: an efficient exact algorithm for the (ℓ, d) - motif finding problem, (2011).

[3] Ying-Jer Liao, Chang-Biau Yang and Shyue-Horng Shia, Motif Finding in Biological Sequences, Masters' Thesis, National Sun Yat-sen University, 2003.

[4] G. D. Stormo, DNA binding sites: representation and discovery, *Bioinformatics*, vol. 16, 2000, pp. 16-23.

[5] S. E. Halford and J. F. Marko, How do site-specific DNA-binding proteins find their targets? *Nucleic Acids Research*, vol. 32, 2004, pp. 3040-52.

[6] D. J. Galas and A. Schmitz, DNAse footprinting: a simple method for the detection of protein-DNA binding specificity, *Nucleic Acids Research*, vol. 5, 1978, pp. 3157-70.

[7] M. M. Garner and A. Revzin, A gel electrophoresis method for quantifying the binding of proteins to specific DNA regions: application to components of the Escherichia coli lactose operon regulatory system, *Nucleic Acids Research*, vol. 9, 1981, pp. 3047-60.

[8] M. Dorigo, V.Maniezzo, and A. Colorni, The ant system: Optimization by a colony of cooperating agents, *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, Vol. 26, No. 1, pp. 29–42, 1996.

[9] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wootton, Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment, *Science*, Vol. 262, pp. 208–214, Oct. 1993.

[10] Xin Chen and Tao Jiang, An improved Gibbs sampling method for motif discovery via sequence weighting, Comput Syst Bioinformatics Conf. 2006:239-47.

[11] Rafael S. Parpinelli, Heitor S. Lopes, and Alex A. Freitas, Data Mining with an Ant Colony Optimization Algorithm.

[12] S. Goss, S. Aron, J. L. Deneubourg, and J. M. Pasteels, self-organized Shortcuts in the Argentine Ant, Unit of Behavioural Ecology, C.P. 231, Universit6 Libre de Bruxelles, B- 1050 Bruxelles.

[13] G. D. Caro and M. Dorigo, Antnet: Distributed stigmergetic control for communications networks, *Journal of Artificial Intelligence Research (JAIR)*, Vol. 9, pp. 317–365, Dec. 1998.

[14] E. Rocke and M. Tompa, An algorithm for finding novel gapped motifs in DNA sequences, *Proceedings of the Second Annual International Conference on ComputationalMolecular Biology*, pp. 228–233,Mar. 1998.

[15] D. S. Latchman, Transcription factors: an overview, *The International Journal of Biochemistry & Cell Biology*, vol. 29, 1997, pp. 1305-12.

[16] Modan K Das, and Ho-Kwok Dai, A survey of DNA motif finding algorithms, *BMC Bioinformatics* 2007, 8(Suppl 7):S21

[17] *www.**ncbi**.nlm.nih.gov*