Motif Refinement using Hybrid Expectation Maximization based Neighborhood profile Search

Chandan K. Reddy
School of Electrical and
Computer Engineering
Cornell University
Ithaca, NY - 14853

Yao-Chung Weng
School of Electrical and
Computer Engineering
Cornell University
Ithaca, NY - 14853

Hsiao-Dong Chiang
School of Electrical and
Computer Engineering
Cornell University
Ithaca, NY - 14853

ABSTRACT

The main goal of the motif finding problem is to detect novel, over-represented unknown signals in a set of sequences (for eg. transcription factor binding sites in a genome). Most widely used algorithms for finding motifs obtain a generative probabilistic representation of these over-represented signals and try to discover profiles that maximize the information content score. Although these profiles form a very powerful representation of the signals, the major difficulty arises from the fact that the best motif corresponds to the global maximum of a non-convex continuous function. Popular algorithms like Expectation Maximization (EM) and Gibbs sampling tend to be very sensitive to the initial guesses and are known to converge to the nearest local maximum very quickly. In order to improve the quality of the results, EM is used with multiple random starts or any other powerful stochastic global methods that might yield promising initial guesses (like projection algorithms). Global methods do not necessarily give initial guesses in the convergence region of the best local maximum but rather suggest that a promising solution is in the neighborhood region. In this paper, we introduce a novel optimization framework that searches the neighborhood regions of the initial alignments in a systematic manner to explore the multiple local optimal solutions. This effective search is achieved by transforming the original optimization problem into its corresponding dynamical system and estimating the practical stability boundary of the local maximum. Our results show that the popularly used EM algorithm often converges to sub-optimal solutions which can be significantly improved by the proposed neighborhood profile search. Based on experiments using both synthetic and real datasets, our method demonstrates significant improvements in the information content scores of the probabilistic models. The proposed method also gives the flexibility in using different local solvers and global methods that work well for some specific datasets.

Corresponding Author - Chandan K. Reddy - Email : ckr6@cornell.edu

General Terms

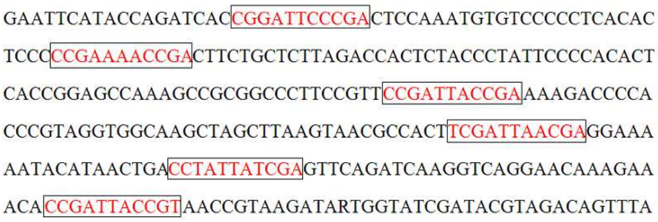
Bioinformatics, Data mining, Dynamical systems, Unsupervised learning.

Keywords

motif finding, global maximum, projection algorithms, expectation maximization, motif refinement

1. INTRODUCTION

Recent developments in DNA sequencing have allowed biologists to obtain complete genomes for several species. However, knowing the sequence does not imply the understanding of how these genes interact and regulate one another within the genome. Many transcription factor binding sites are usually highly conserved throughout the sequences and discovering the location of such binding sites plays an important role in the inference of the gene interaction and gene regulation. A *motif* is a sequence of DNA which manifests itself repetitively throughout genomic sequences. The length of the motif in each occurrence may not be the same as all of the other occurrences, although in general the occurrences must have roughly the same length. The motif challenge problem [19] that is being considered in this paper is described as follows: Given N sequences with t_i being the length of the i^{th} sequence, the goal of the motif finding problem is to locate all occurrences of the l-length motif which is within a distance of d mutations in each of the t sequences (see fig. 1). More details about the complexity of the motif finding problem is given in [18]. A detailed assessment of different motif finding algorithms has been published recently in [26].



```
GAATTCATACCAGATCACCGGATTCCCGACTCCAAATGTGTCCCCCTCACAC
TCCCCCGAAAACCGACTTCTGCTCTTAGACCACTCTACCCTATTCCCCACACT
CACCGGAGCCAAAGCCGCGGCCCTTCCGTTCCGATTACCGAAAAGACCCCA
CCCGTAGGTGGCAAGCTAGCTTAAGTAACGCCACTTCGATTAACGAGGAAA
AATACATAACTGACCTATTATCGAGTTCAGATCAAGGTCAGGAACAAAGAA
ACACCGATTACCGTAACCGTAAGATARTGGTATCGATACGTAGACAGTTTA
```

Figure 1: Synthetic DNA sequences containing some instance of the pattern 'CCGATTACCGA' with a maximum number of 2 mutations. The motifs in each sequence are highlighted in the box. We have a (11,2) motif where 11 is the length of the motif and 2 is the number of mutations allowed.

Although there are several variations of the motif finding algorithms, the problem discussed in this paper is defined as follows: without any previous knowledge about the consensus pattern, discover all instances (alignment positions) of the motifs and then recover the final pattern to which all these instances are within a given number of mutations. Inspite of the significant literature on the motif finding problem, relatively few researchers have exploited the probabilistic models used for motif refinement [16],[1]. More details on the estimates of the hardness of this problem without any complex information like overlapping motifs and background distribution is shown in [27].

In this paper, we provide a novel optimization framework for refining motifs using systematic subspace exploration and neighborhood search techniques. This paper is organized as follows: Section 2 gives some relevant background about the existing approaches used for finding motifs. Section 3 describes the problem formulation in detail. Section 4 discusses our new framework and Section 5 details our implementation. Section 6 gives the experimental results of our algorithm on synthetic and real datasets. Finally, Section 7 concludes our discussion with future research directions.

2. RELEVANT BACKGROUND

Existing approaches used to solve the motif finding problem can be classified into two main categories [10]. The first group of algorithms utilizes a generative probabilistic representation of the nucleotide positions to discover a consensus DNA pattern that maximizes information content score. In this approach, the original problem of finding the best consensus pattern is formulated into finding the global maximum of a continuous non-convex function. The main advantage of this approach is that profiles generated are highly representative of the signals being determined [7]. The disadvantage, however, is that the determination of the "best" motif cannot be guaranteed and is often very difficult since finding global maximum of any continuous non-convex function is a challenging problem. Current algorithms converge to the nearest local optimum instead of the global solution. Gibbs sampling [15], Expectation-Maximization [1], greedy CONSENSUS algorithm [13] and HMM based methods [8] belong to this category.

The second group uses patterns with 'mismatch representation' which defines a signal to be a consensus pattern and allows up to a certain number of mismatches to occur in each instance of the pattern. The goal of these algorithms is to recover the consensus pattern with the highest number of instances. These methods view the representation of the signals as discrete and the main advantage to these algorithms is that they can guarantee the highest scoring pattern to be the global optimum for any scoring function. The disadvantage, however, is that consensus patterns are not as expressive of the DNA signal as the profile representations. Recent approaches within this framework include Projection methods [4; 22], string based [19], Pattern-Branching [21], MULTIPROFILER [14], suffix trees [24] and other branch and bound approaches [11; 10].

A hybrid approach could potentially combine the expressiveness of the profile representation with convergence guarantees of the consensus pattern. An example of a hybrid approach is the Random Projection [4] algorithm followed by Expectation-Maximization [1]. It uses a global solver to obtain promising alignments in the discrete pattern space followed by further local solver refinements in continuous space[2; 25]. Currently, not many algorithms take complete advantage of the combined discrete and continuous space search [4; 10; 22]. In this paper, the profile representation of the motif is emphasize and a new hybrid algorithm is developed to escape out of the local maximum of the likelihood surface.

The main research concerns that motivated the new hybrid algorithm proposed in this paper are :

- Motif refinement stage is vital and popularly used by many pattern based algorithms (like PROJECTION, MITRA etc) that try to find optimal motifs.
- Traditional Expectation Maximization algorithm used in the context of motif finding converges very quickly to the nearest local optimal solution (within 5-8 iterations).
- There are many other promising local optimal solutions in the close vicinity of the profiles obtained from the global methods.

In spite of the importance of obtaining a globally optimal solution in the context of motif finding, not much work has been done in the direction of finding such solutions [28; 12]. There had been several attempts for escaping out of the local optimal solution to find better solutions in other machine learning [9] and optimization [5] related problems. Most of these methods are stochastic in nature and usually rely on perturbing either the data or the hypothesis. These stochastic perturbation algorithms are ineffecient because they sometimes miss a neighborhood solution or obtain an already existing solution. To avoid these problems, we introduce a novel optimization framework that has a better chance of avoiding sub-optimal solutions. It systematically escapes out of the convergence region of a local maximum to explore the existence of other neighborhood local maxima. Our method is primarily based on some fundamental principles of finding an exit points on the stability boundary of a nonlinear continuous function. The underlying theoretical details of our method are described in [6; 17].

3. PRELIMINARIES

Before discussing the details of our method, we describe our problem formulation and the details about the EM algorithm in the context of motif finding problem. We also describe some details about the dynamical system of the log-likelihood function which enables us to search the nearby local optimal solutions.

3.1 Problem Formulation

Some promising initial alignments are obtained by applying projection methods or random starts on the entire dataset. These initial alignments are then converted into profile representation.

Let t be the total number of sequences and n be the average length of each sequence. Let $S = \{S_1, S_2...S_t\}$ be the set of t sequences. Let $P = \{P_1, P_2...P_t\}$ be the set of initial alignments. l is the length of the consensus pattern. For further discussion, we use the following variables

$$\begin{aligned} i &= 1 \;...\; t & \quad & \text{\% for } t \text{ sequences} \\ k &= 1 \;...\; l & \quad & \text{\% for } l\text{-mers} \end{aligned}$$

Table 1: A count of nucleotides A, T, G, C at each position $K = 1..l$ in all the sequences of the data set. $K = 0$ denotes the background count.

j	$k=0$	$k=1$	$k=2$	$K=3$	$k=4$	...	$k=l$
A	$C_{0,1}$	$C_{1,1}$	$C_{2,1}$	$C_{3,1}$	$C_{4,1}$	...	$C_{l,1}$
T	$C_{0,2}$	$C_{1,2}$	$C_{2,2}$	$C_{3,2}$	$C_{4,2}$	...	$C_{l,2}$
G	$C_{0,3}$	$C_{1,3}$	$C_{2,3}$	$C_{3,3}$	$C_{4,3}$	...	$C_{l,3}$
C	$C_{0,4}$	$C_{1,4}$	$C_{2,4}$	$C_{3,4}$	$C_{4,4}$	...	$C_{l,4}$

$$j \in \{A, T, G, C\} \quad \text{\% for each nucleotide}$$

The count matrix can be constructed from the given alignments as shown in Table 1. We define $C_{0,j}$ to be the non-position specific background count of each nucleotide in all of the sequences where $j \in \{A, T, C, G\}$ is the running total of nucleotides occurring in each of the l positions. Similarly, $C_{k,j}$ is the count of each nucleotide in the k^{th} position (of the $l - MER$) in all the P alignments.

$$Q_{0,j} = \frac{C_{0,j}}{\sum_j C_{0,j}} \tag{1}$$

$$Q_{k,j} = \frac{C_{k,j} + b_j}{t + \sum_j b_j} \tag{2}$$

Equation (1) shows the background frequency of each nucleotide where b_j is known as the Laplacian or Bayesian correction and is equal to $d * Q_{0,j}$ where d is some constant usually set to unity. Equation (2) gives the weight assigned to the type of nucleotide at the k^{th} position of the motif.
A Position Specific Scoring Matrix (PSSM) can be constructed from one set of instances in a given set of t sequences. From (1) and (2), it is obvious that the following relationship holds:

$$\sum_{j \in \{A,T,G,C\}} Q_{k,j} = 1 \qquad \forall k = 0, 1, 2, ...l \tag{3}$$

From equation (3), for a given k value, each Q can be represented in terms of the other 3 variables. Since the length of the motif is l, the final objective function (i.e. the information content score) would contain $3l$ independent variables[1]. To obtain the score, every possible $l - MER$ in each of the t sequences must be examined. This is done so by multiplying the respective $Q_{i,j}/Q_{0,j}$ dictated by the nucleotides and their respective positions within the $l - MER$. Only the highest scoring $l - MER$ in each sequence is noted and kept as part of the alignment. The total score is the sum of all the best scores in each sequence.

$$\begin{aligned} A(Q) &= \sum_{i=1}^{t} log(A)_i = \sum_{i=1}^{t} log\left(\prod_{k=1}^{l} \frac{Q_{k,j}}{Q_b}\right)_i \\ &= \sum_{i=1}^{t}\sum_{k=1}^{l} log(Q'_{k,j})_i \end{aligned} \tag{4}$$

$Q'_{k,j}$ is the ratio of the nucleotide probability to the corresponding background probability, i.e. $Q_{k,j}/Q_b$. $Log(A)_i$ is the score at each individual i^{th} sequence where t is the total number of sequences. In equation (4), we see that A is composed of the product of the weights for each individual position k. $A(Q)$ is the non-convex $3l$ dimensional continuous function for which the global maximum corresponds to the best possible motif in the dataset. EM refinement that is done at the end of the combinatorial approaches has the main disadvantage that it converges to a local optimal solution [3]. Our method improves the refinement procedure by understanding the details about the stability boundaries and trying to escape out of the convergence region of the EM algorithm.

3.2 Hessian Computation and Dynamical System for the Scoring Function

In order to present our algorithm, we have defined the dynamical system corresponding to the log-likelihood function and the PSSM. The key contribution of the paper is the development of this nonlinear dynamical system which will enable us to realize the dynamic and geometric nature of the likelihood surface. We construct the following *gradient system* in order to locate critical points of the objective function (4):

$$\dot{Q}(t) = -\nabla A(Q) \tag{5}$$

One can realize that this transformation preserves all the critical points [6]. Now, we will describe the construction of the gradient system and the Hessian in detail. In order to reduce the dominance of one variable over the other, the values of the each of the nucleotides that belong to the consensus pattern at the position k will be represented in terms of the other three nucleotides in that particular column. This will also minimize the dominance of the eigen vector directions when the Hessian is obtained. The variables of the scoring function are transformed into new variables described in Table 2.

$$A(Q) = \sum_{i=1}^{t}\sum_{k=1}^{l} log\ f_{ik}(w_{3k-2}, w_{3k-1}, w_{3k})_i \tag{6}$$

where

$$f_{ik} = \begin{cases} 1 - (w_{3k-2}, w_{3k-1}, w_{3k}) & \text{if } P_{ik} = C_k \\ w_{3k-2}\ or\ w_{3k-1}\ or\ w_{3k} & \text{elsewhere} \end{cases} \tag{7}$$

The first derivative of the scoring function is a one dimensional vector with $3l$ elements.

$$\nabla A = \left[\ \frac{\partial A}{\partial w_1}\ \ \frac{\partial A}{\partial w_2}\ \ \frac{\partial A}{\partial w_3}\ \ \ldots.\ \ \frac{\partial A}{\partial w_{3l}}\ \right]^T \tag{8}$$

[1] Although, there are $4l$ variables in total, because of the constraints obtained from (3), the parameter space will contain only $3l$ independent variables. Thus, the constraints help in reducing the dimensionality of the search problem.

Table 2: A count of nucleotides $j \in \{A, T, G, C\}$ at each position $k = 1..l$ in all the sequences of the data set. C_k is the k^{th} nucleotide of the consensus pattern which represents the nucleotide with the highest value in that column. Let the consensus pattern be GACT...G and b_j indicates the background.

j	$k=b$	$k=1$	$k=2$	$K=3$	$k=4$	...	$k=l$
A	b_A	w_1	C_2	w_7	w_{10}	...	w_{3l-2}
T	b_T	w_2	w_4	w_8	C_4	...	w_{3l-1}
G	b_G	C_1	w_5	w_9	w_{11}	...	C_l
C	b_C	w_3	w_6	C_3	w_{12}	...	w_{3l}

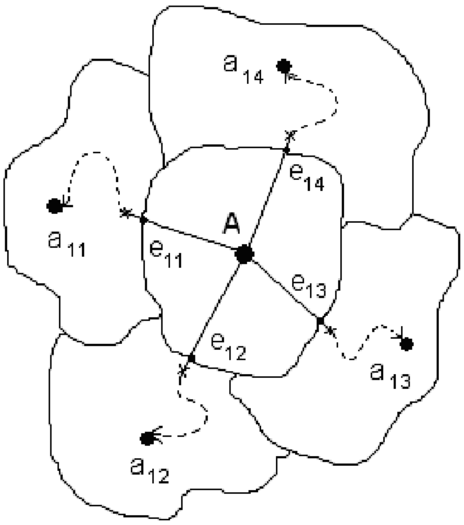


Figure 2: Diagram illustrates the exit point method of escaping from the original solution (A) to the neighborhood local optimal solutions (a_{1i}) through the corresponding exit points (e_{1i}). The dotted lines indicate the local convergence of the EM algorithm.

and each partial derivative is given by

$$\frac{\partial A}{\partial w_p} = \sum_{i=1}^{t} \frac{\frac{\partial f_{ip}}{\partial w_p}}{f_{ik}(w_{3k-2}, w_{3k-1}, w_{3k})} \tag{9}$$

$$\forall p = 1, 2 \ ... \ 3l \ \ and \ \ k = round(p/3) + 1$$

The Hessian $\nabla^2 A$ is a block diagonal matrix of block size 3X3. For a given sequence, the entries of the 3X3 block will be the same if that nucleotide belongs to the consensus pattern (C_k). The gradient system is mainly obtained for enabling us to identify the stability boundaries and stability regions on the likelihood surface. The theoretical details about these concepts are published elsewhere [6]. Stability region of each local maximum is an approximate convergence zone of the EM algorithm. If we can identify all the saddle points on the stability boundary of a given local maximum, then we will be able to find all the tier-1 local maxima. However, finding all saddle points is computationally intractable and hence we have adopted a heuristic by generating the eigen vector directions of the PSSM at the local maximum. The next section details out our approach and explains the different phases of our algorithm.

4. NOVEL FRAMEWORK

Our framework consists of three phases. The first phase is the *global phase*, in which the promising solutions in the entire search space are obtained. The second phase is the *refinement phase* where a local method is applied to the solutions obtained in the previous phase in order to refine the profiles. The third phase is the *exit phase*; the exit points are computed and the Tier-1 and Tier-2 solutions are systematically explored.

In the global phase, a branch and bound search is performed on the entire dataset. All the profiles that do not meet a certain threshold (in terms of a given scoring function) are eliminated in this phase. The promising patterns obtained are transformed into profiles and local improvements are made to these profiles in the refinement phase. The consensus pattern is obtained from each nucleotide that corresponds to the largest value in each column of the PSSM. The $3l$ variables chosen are the nucleotides that correspond to those that are not present in the consensus pattern. Because of the probability constraints discussed in the previous section, the largest weight can be represented in terms of the other three variables.

To solve (4), current algorithms begin at random initial alignment positions and attempt to converge to an alignment of $l - MER$s in all the sequences that maximize the objective function. In other words, the $l - MER$ whose $log(A)_i$ is the highest (with a given PSSM) is noted in every sequence as part of the current alignment. During the maximization of $A(Q)$ function, the probability weight matrix and hence the corresponding alignments of $l - MER$s are updated. This will occur iteratively until the PSSM converges to the locally optimal solution. The consensus pattern is obtained from the largest weight nucleotide in each position (column) of the PSSM. This converged PSSM and the set of alignments correspond to a local optimal solution. To escape out of this local optimal solution, our approach requires the computation of a Hessian matrix (i.e. the matrix of second derivatives) of dimension $(3l)^2$ and the $3l$ eigenvectors of the Hessian. The Hessian $\nabla^2 A$ is a block diagonal

Input: Local Maximum (A).
Output: Best Local Maximum in the neighborhood region.
Algorithm:
Step 1: Construct the PSSM for the alignments corresponding to the local maximum (A) using Eqs. 1 and 2.
Step 2: Calculate the eigen vectors of the Hessian matrix for this PSSM.
Step 3: Find exit points (e_{1i}) on the practical stability boundary along each eigen vector direction.
Step 4: For each of the exit points, the corresponding Tier-1 local maxima are obtained (a_{1i}) by applying the EM algorithm after the ascent step.
Step 5: Repeat this process for promising tier-1 solutions to obtain Tier-2 (a_{2j}) local maxima.
Step 6: Return the solution that gives the maximum information content score of $\{A, \ a_{1i}, \ a_{2j}\}$.

Figure 3: The Exit phase where the neighborhood of the original solution is explored in a systematic manner.

matrix of block size 3X3. For a given sequence, the entries of the 3X3 block will be the same if that nucleotide belongs to the consensus pattern (C_k). The main reasons for choosing the eigenvectors of the Hessian as search directions are:

- Computing eigen vectors of the Hessian is related to finding the directions with extreme values of the second derivatives, i.e., directions of extreme normalto-isosurface change.
- Eigen vectors of the Hessian will form the basis vectors for the search directions. Any other search direction can be obtained by a linear combination of these basis directions.
- This will make our algorithm deterministic since the eigen vector directions are always unique.

The value of the objective function is evaluated along these eigen vector directions with some small step size increments. Since the starting position is a local optimal solution of a function that is being maximized, the function value during the initial few steps will reduce. Since the Hessian is obtained only once during the entire procedure, it is more efficient compared to Newton's method where an approximate Hessian is obtained for every iteration. After a certain number of step evaluations, there might be an increase in the value indicating that the current point is out of the convergence region of the local maximum. The point along this direction where the $A(Q)$ has the lowest value is called the *exit point*. Once the exit points are computed along each eigenvector direction, the local maximum in the other region is obtained by applying local method with these new points as initial guesses. This procedure is clearly shown in Fig 4. To ascertain that the new initial guess is in a different convergence region from the original, the objective function value is evaluated even after its increase. The descent stage indicates the function evaluation along a particular eigen vector direction. Applying local method at the exit point might give the original local maximum. The ascent stage is used to ensure that the new guess is in a different convergence zone. Hence, given the best local maximum obtained using any current local methods, this framework allows us to systematically escape out of the local maximum to explore surrounding local maxima.

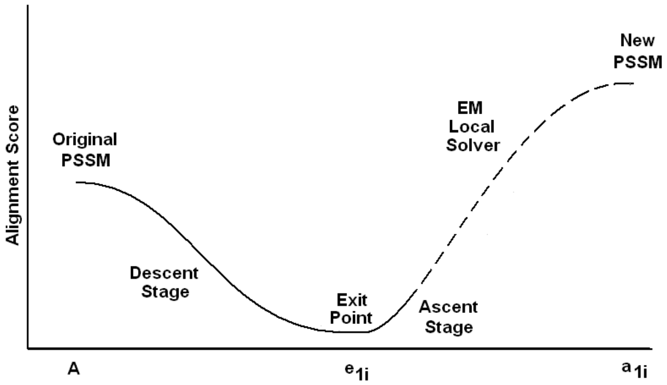


Figure 4: A summary of escaping out of the local optimum to the neighborhood local optimum. Observe the corresponding trend of $A(Q)$ at each step.

Input: The DNA sequences, length of the motif (l), Maximum Number of Mutations (d)
Output: Motif (s)
Algorithm:
Step 1: Given the sequences, apply random projection algorithm to obtain different set of alignments.
Step 2: Choose the promising buckets and apply EM algorithm to refine these alignments.
Step 3: Apply the exit point method to obtain nearby promising local optimal solutions.
Step 4: Report the consensus pattern that corresponds to the best alignments and their corresponding PSSM.

Figure 5: The complete algorithm

This new framework can be treated as a hybrid approach between global method and the local method. The approach differs from traditional local methods by computing multiple local solutions in the neighborhood region in a systematic manner. It differs from global methods by working completely in profile space and searching a subspace efficiently in a deterministic manner. For a given non-convex function, there is a massive number of convergence regions that are very close to each other and are separated from one another in the form of different basins of attraction. These basins are effectively modeled by the concepts of stability regions.

5. IMPLEMENTATION DETAILS

Our program is implemented in Red Hat Linux version 9 and runs on a Pentium IV 2.8 GHz machine. The core algorithm that we implemented is XP_EM described in Algorithm 1. XP_EM obtains the initial alignments and the original data sequences along with the length of the motif. It returns the best motif that is obtained in the neighboring region of the sequences. This procedure constructs the PSSM, performs EM refinement, and then computes the Tier-1 and Tier-2 solutions by calling the procedure $Next\_Tier$. The Eigen vectors of the Hessian were computed using the code obtained from [20]. $Next\_Tier$ takes a PSSM as input and computes an array of PSSMs corresponding to the next tier local maxima using the exit point methodology.

Algorithm 1 Motif $XP\_EM(init\_aligns, seqs, l)$

```
$PSSM = Construct\_PSSM(init\_aligns)$
$New\_PSSM = Apply\_EM(PSSM, seqs)$
$TIER1 = Next\_Tier(seqs, New\_PSSM, l)$
for $i = 1$ to $3l$ do
  if $TIER1[i] <> zeros(4l)$ then
    $TIER2[i][\,] = Next\_Tier(seqs, TIER1[i], l)$
  end if
end for
Return $best(PSSM, TIER1, TIER2)$
```

Given a set of initial alignments, Algorithm 1 will find the best possible motif in the neighborhood space of the profiles. Initially, a PSSM is computed using *construct_PSSM* from the given alignments. The procedure *Apply_EM* will return a new PSSM that corresponds to the alignments obtained after the Expectation Maximization algorithm is applied to the initial PSSM. The details of the procedure $Next\_Tier$

are given in Algorithm 2. From a given local solution (or PSSM), $Next\_Tier$ will compute all the $3l$ new $PSSMs$ in the neighborhood of the given local optimal solution. The second tier patterns are obtained by calling the $Next\_Tier$ from every first tier solutions [2]. Finally, the pattern with the highest score amongst the original PSSM, Tier1 and Tier2 is returned.

Algorithm 2 PSSMs[ ] $Next\_Tier(seqs, PSSM, l)$

```
$Score = eval(PSSM)$
$Hess = Construct\_Hessian(PSSM)$
$Eig[\ ] = Compute\_EigVec(Hess)$
$MAX\_Iter = 100$
for $k = 1$ to $3l$ do
    $PSSMs[k] = PSSM$      $Count = 0$
    $Old\_Score = Score$      $ep\_reached = FALSE$
    while (! $ep\_reached$) && ($Count < MAX\_Iter$) do
        $PSSMs[k] = update(PSSMs[k], Eig[k], step)$
        $Count = Count + 1$
        $New\_Score = eval(PSSMs[k])$
        if ($New\_Score > Old\_Score$) then
            $ep\_reached = TRUE$
        end if
        $Old\_Score = New\_Score$
    end while
    if $count < MAX\_Iter$ then
        $PSSMs[k] = update(PSSMs[k], Eig[k], ASC)$
        $PSSMs[k] = Apply\_EM(PSSMs[k], Seqs)$
    else
        $PSSMs[k] = zeros(4l)$
    end if
end for
Return $PSSMs[\ ]$
```

The procedure $Next\_Tier$ takes a PSSM and computes an array of PSSMs that corresponds to the next tier local optimal solutions. It applies the Exit-point methodology to compute the next tier solution. The procedure $eval$ evaluates the scoring function for the PSSM using (4). The procedures $Construct\_Hessian$ and $Compute\_EigVec$ computes the Hessian matrix and the eigen vectors respectively. MAX_iter indicates the maximum number of uphill evaluations that are required along each of the eigen vector directions. The neighborhood PSSMs will be stored in the variable $PSSMs[\ ]$. The original PSSM is updated with a small step until an exit point is reached or the number of iterations exceed MAX_Iter value. If the exit point is reached along a particular direction, some more iterations are made to guarantee that the PSSM exists in a different stability region and entered a new one. The EM algorithm is then used during this ascent stage to obtain a new PSSM [3].

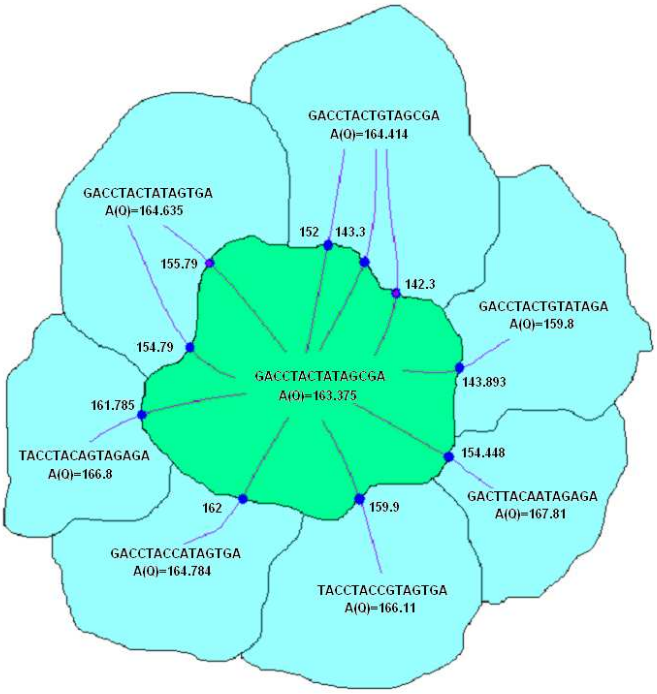


Figure 6: 2-D illustration of first tier improvements in a $3l$ dimensional objective function. The original local maximum has a score of 163.375. The various Tier-1 solutions and plotted and the one with highest score (167.81) is chosen.

The initial alignments are converted into profile space and a PSSM is constructed. The PSSM is updated (using the EM algorithm) until the alignments converge to a local optimal solution. The Exit-point methodology is then employed to escape out of this local optimal solution to compute nearby first tier local optimal solutions. This process is then repeated on promising first tier solutions to obtain second tier solutions. As shown in Fig. 6, from the original local optimal solution, various exit points and correspondingly the new local optimal solutions are computed along each Eigen vector direction. Sometimes two directions might yield the same local optimal solution. This can be avoided by computing the saddle point corresponding to the exit point on the stability boundary [23]. There can be many exit points, but there will be a unique saddle point corresponding to the new local minimum. However, in high dimensional problems, it is not very efficient to compute the saddle points and hence, we chose to compute the exit points. For computational efficiency, the Exit-point approach is applied to only promising initial alignments (i.e. random starts with higher Information Content score). Therefore, a threshold $A(Q)$ score is determined by the average of the best three first tier scores just after 10-15 random starts; any current and future first tier solution with score greater than the threshold is considered for further analysis. Additional random starts are carried out in order to aggregate at least ten first tier solutions. Exit-point is repeated on all first tier solutions above a certain threshold to obtain second tier solutions.

6. EXPERIMENTAL RESULTS

Experiments were performed on both synthetic data and real data. Two different methods were used in the global phase:

[2] New PSSMs might not be obtained for certain search directions. In those cases, a zero vector of length $4l$ is returned back. Only those new PSSMs which do not have this value will be used for any further processing.

[3] For completeness, the entire algorithm has been shown in this section. However, during the implementation, several heuristics have been applied to reduce the running time of the algorithm. For example, if the first tier solution is not very promising, it will not be considered for obtaining the corresponding second tier solutions.

random starts and random projection. The main purpose of this paper is not to demonstrate that our algorithm can outperform the existing motif finding algorithms. Rather, the main work here focusses on improving the results that are obtained from other efficient algorithms. We have chosen to demonstrate the performance of our algorithm on the results obtained from the random projection method which is a powerful global method that has outperformed other traditional motif finding approaches like MEME, Gibbs sampling, WINNOWER, SP-STAR etc [4]. Since the comparison results were already published, we mainly focus on the performance improvements of our algorithm compared to the random projection algorithm. For random starts experiment, a total of N random numbers each value between 1 and $(t - l + 1)$ that corresponds to random initial starting alignments are generated. Let m be the number of independent trials required to obtain the motifs.

6.1 Synthetic Datasets

The synthetic datasets were generated using the procedure described in [19]. The value of $m = 1$ is chosen to demonstrate the efficiency of our approach. This corresponds to one full random projection + EM cycle. We compared the performance coefficient (PC) which gives the measure of the average performance of our implementation to that of Random Projection. The PC is given by :

$$PC = \frac{|K \cap P|}{|K \cup P|} \tag{10}$$

where K is the set of the residue positions of the planted motif instances, and P is the corresponding set of positions predicted by the algorithm. Table 4 gives an overview of the performance of our method compared to the random projection algorithm on the (l,d) motif problem for different l and d values.

Our results also show that by branching out and discovering multiple local optimal solutions one need not use higher m values. A higher m value corresponds to more computational time because projecting the l-mers into k-sized buckets is a time consuming task. Using our approach, we can replace the need for randomly projecting l-mers repeatedly in an effort to converge to a global optimum by deterministically and systematically searching the solution space modeled by our dynamical system and improving the quality of the existing solutions. The improvements of our algorithm are clearly shown in Table 4. We can see that there is a significant improvement for higher length motifs.

Table 4: The results of performance coefficient with $m = 1$ on synthetically generated sequences. The scores are not normalized and the perfect score is 20 since there are 20 sequences.

Motif (l,d)	PC obtained using Random Projection	PC obtained using Exit-point method
(11,2)	20	20
(15,4)	14.875	17
(20,6)	12.667	18

Fig. 6 shows the tier-1 solutions obtained from a given consensus pattern. Since the exit points are being used instead of saddle points, it might sometimes find the same local optimal solution obtained before. As seen from the figure, the tier-1 solutions does not have to be different from the original pattern in just one nucleotide position. Also, the function value at the exit points is much higher than the original value.

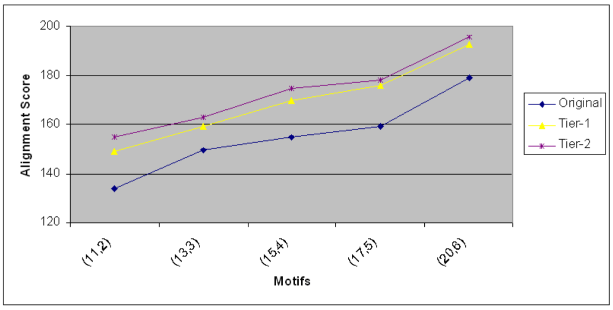


Figure 7: The average scores with the corresponding first tier and second tier improvements on synthetic data using the random starts with Exit-point approach with different (l,d) motifs.

As opposed to stochastic processes like mutations in genetic algorithms, our approach reduces the stochastic nature and tries to obtain multiple local optimal solutions in the neighborhood systematically. Fig. 7 shows the performance of the Exit-point approach on synthetic data for different (l,d) motifs. The average scores of the best ten solutions obtained from random starts and their corresponding improvements in tier-1 and tier-2 are reported. One can see that the improvements become more prominent for larger length motifs. Table 3 shows the best and worst of these top ten random starts along with the consensus pattern and the alignment scores.

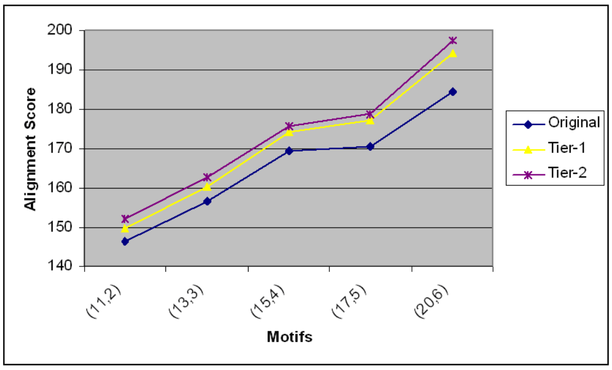


Figure 8: The average scores with the corresponding first tier and second tier improvements on synthetic data using the Random Projection with Exit-point approach with different (l,d) motifs.

With a few modifications, more experiments were conducted using the Random Projection method. The Random Projection will eliminate non-promising regions in the search space and gives a number of promising sets of initial patterns. EM refinement is applied to those promising initial patterns with higher score. Due to the robustness of the results, the Exit-point method is employed only on the top five local optima. Exit-point is again repeated on the top scoring first tier so-

Table 3: The consensus patterns and their corresponding scores of the original local optimal solution obtained from multiple random starts on the synthetics data. The best first tier and the second tier optimal patterns and their corresponding scores are also reported.

(l,d)	Initial Pattern	Score	First Tier Pattern	Score	Second Tier Pattern	Score
(11,2)	AACGGTCGCAG	125.1	CCCGGTCGCTG	147.1	CCCGGGAGCTG	153.3
(11,2)	ATACCAGTTAC	145.7	ATACCAGTTTC	151.3	ATACCAGGGTC	153.6
(13,3)	CTACGGTCGTCTT	142.6	CCACGGTTGTCTC	157.8	CCTCGGGTTTGTC	158.7
(13,3)	GACGCTAGGGGGT	158.3	GAGGCTGGGCAGT	161.7	GACCTTGGGTATT	165.8
(15,4)	CCGAAAAGAGTCCGA	147.5	CCGCAATGACTGGGT	169.1	CCGAAAGGACTGCGT	176.2
(15,4)	TGGGTGATGCCTATG	164.6	TGGGTGATGCCTATG	166.7	TGAGAGATGCCTATG	170.4
(17,5)	TTGTAGCAAAGGCTAAA	143.3	CAGTAGCAAAGACTACC	173.3	CAGTAGCAAAGACTTCC	175.8
(17,5)	ATCGCGAAAGGTTGTGG	174.1	ATCGCGAAAGGATGTGG	176.7	ATTGCGAAAGAATGTGG	178.3
(20,6)	CTGGTGATTGAGATCATCAT	165.9	CAGATGGTTGAGATCACCTT	186.9	CATTTAGCTGAGTTCACCTT	194.9
(20,6)	GGTCACTTAGTGGCGCCATG	216.3	GGTCACTTAGTGGCGCCATG	218.8	CGTCACTTAGTCGCGCCATG	219.7

lutions to arrive at the second tier solutions. Fig. 8 shows the average alignment scores of the best random projection alignments and their corresponding improvements in tier-1 and tier-2 are reported. In general, the improvement in the first tier solution is more significant than the improvements in the second tier solutions.

6.2 Real Datasets

Table 5 shows the results of the Exit-point methodology on real biological sequences. We have chosen $l = 20$ and $d = 2$. 't' indicates the number of sequences in the real data. The m value reported is the approximate average number of full cycles required to obtain the motif. For the biological samples taken from [4; 21], the value of m is the average number of full LSH cycles it would take the original algorithm to discover the motif. The values for all other parameters (like projection size $k = 7$ and threshold s=4) are chosen to be the same as those used in the Random projection paper [4]. All the motifs were recovered with $m = 1$ using the Exit-point strategy. Without the exit point strategy, the random projection algorithm needed multiple LSH cycles inorder to retrieve the original motifs. This clearly elucidates the fact that, we need to use global methods to certain extent and combine them with refined local heuristics in order to obtain better efficiency. Running one LSH cycle is much more time consuming compared to the exit-point strategy. The main advantage of our strategy comes with the deterministic nature of the algorithm as opposed to the stochastic version (as seen in random projection). This clearly indicates the efficiency of the newly proposed method on real biological samples.

7. CONCLUDING DISCUSSION

The Exit-point framework proposed in this paper broadens the search region for obtaining improved solution that can potentially corresponds to a better motif. In most of the profile based algorithms, EM is used to obtain the nearest local optimum from a given starting point. In our approach, we consider the boundaries of these convergence regions and find the surrounding local optimal solution based on the theory of stability regions. We have shown in both real and synthetic data sets that beginning from the EM converged solution, the Exit-point approach is capable of searching in the neighborhood regions for another solution with an improved information content score. This will often translate to finding a pattern with less hamming distance from the resulting alignments in each sequence. Our approach has shown improvements in the score on all datasets that it was tested on. One of the primary advantages of our method is that it can be used with different global and local methods. The main contribution of our work is to demonstrate the capability of this hybrid expectation maximization algorithm in the context of the motif finding problem. We can potentially use any global method and improve its results efficiently.

From our results, we see that motif refinement stage in the motif finding problem plays a vital role and can yield accurate results more efficiently in terms of computational costs. We would like to continue our work by combining other global methods that are available in the literature with existing local solvers like EM or GibbsDNA that work in continuous space. Implementing the Exit-point method as an intermediate between the global and local solver provides us with a fundamental advantage of choosing different methods to explore the data specific properties in more detail. We will follow the example of [26] and try different combinations of the existing methods to improve the chances of finding more promising patterns.

8. REFERENCES

[1] T. Bailey and C. Elkan. Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *The First International Conference on Intelligent Systems for Molecular Biology*, pages 28–36, 1994.

[2] Y. Barash, G. Bejerano, and N. Friedman. A simple hyper-geometric approach for discovering putative transcription factor binding sites. *Proc. of First International Workshop on Algorithms in Bioinformatics*, 2001.

[3] K. Blekas, D. Fotiadis, and A. Likas. Greedy mixture learning for multiple motif discovery in biological sequences. *Bioinformatics*, 19(5):607–617, 2003.

[4] J. Buhler and M. Tompa. Finding motifs using random projections. *Proceedings of the fifth annual international conference on Research in computational molecular biology*, pages 69–76, 2001.

[5] B. C. Cetin, J. Barhen, and J. W. Burdick. Terminal repeller unconstrained subenergy tunneling (TRUST) for fast global optimization. *Journal of Optimization Theory and Applications*, 77(1):97–126, 1993.

[6] H.D. Chiang and C.C. Chu. A systematic search method for obtaining multiple local optimal solutions of

Table 5: Results of Exit-point methodology on biological samples. The real motifs were obtained in all the six cases using the Exit-point framework.

Sequence	Sample Size	t	Best (20,2) Motif	Reference Motif
E. coli CRP	1890	18	TGTGAAATAGATCACATTTT	TGTGANNNNGNTCACA
preproinsulin	7689	4	GGAAATTGCAGCCTCAGCCC	CCTCAGCCC
DHFR	800	4	CTGCAATTTCGCGCCAAACT	ATTTCNNGCCA
metallothionein	6823	4	CCCTCTGCGCCCGGACCGGT	TGCRCYCGG
c-fos	3695	5	CCATATTAGGACATCTGCGT	CCATATTAGAGACTCT
yeast ECB	5000	5	GTATTTCCCGTTTAGGAAAA	TTTCCCNNTNAGGAAA

nonlinear programming problems. *IEEE Transactions on Circuits and Systems: I Fundamental Theory and Applications*, 43(2):99–109, 1996.

[7] R. Durbin, S.R. Eddy, A. Krogh, and G. Mitchison. *Biological Sequence Analysis : Probabilistic Models of Proteins and Nucleic Acids.* Cambridge University Press, 1999.

[8] S. R. Eddy. Profile hidden markov models. *Bioinformatics*, 14(9):755–763, 1998.

[9] G. Elidan, M. Ninio, N. Friedman, and D. Schuurmans. Data perturbation for escaping local maxima in learning. *In Proceedings of the Eighteenth National Conference on Artificial Intelligence*, pages 132 – 139, 2002.

[10] E. Eskin. From profiles to patterns and back again: A branch and bound algorithm for finding near optimal motif profiles. *Proceedings of the eighth annual international conference on Research in computational molecular biology*, pages 115–124, 2004.

[11] E. Eskin and P. Pevzner. Finding composite regulatory patterns in dna sequences. *Tenth International Conference on Intelligent Systems for Molecular Biology*, pages 354–363, 2002.

[12] G. B. Fogel, D. G. Weekes, G. Varga, E. R. Dow, H. B. Harlow, J. E. Onyia, and C. Su1. Discovery of sequence motifs related to coexpression of genes using evolutionary computation. *Nucleic Acids Research*, 32(13):3826–3835, 2004.

[13] G. Hertz and G. Stormo. Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, 15(7/8):563577, 1999.

[14] U. Keich and P. Pevzner. Finding motifs in the twilight zone. *Bioinformatics*, 18:1374–1381, 2002.

[15] C. Lawrence, S. Altschul, M. Boguski, J. Liu, A. Neuwald, and J. Wootton. Detecting subtule sequence signals: a gibbs sampling strategy for multiple alignment. *Science*, 262:208–214, 1993.

[16] C. E. Lawrence and A. A. Reilly. An expectation maximization (EM) algorithm for the identification and characterization of common sites in unaligned biopolymer sequences. *Proteins: Structure, Function, and Genetics*, 7:41–51, 1990.

[17] J. Lee and H.D. Chiang. A dynamical trajectory-based methodology for systematically computing multiple optimal solutions of general nonlinear programming problems. *IEEE Transactions on Automatic Control*, 49(6):888 – 899, 2004.

[18] P. Pevzner. *Computational Molecular Biology - an algorithmic approach*, chapter Finding Signals in DNA, pages 133–152. MIT Press, 2000.

[19] P. Pevzner and S-H. Sze. Combinatorial approaches to finding subtle signals in DNA sequences. *The Eighth International Conference on Intelligent Systems for Molecular Biology*, pages 269–278, 2000.

[20] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 1992.

[21] A. Price, S. Ramabhadran, and P.A. Pevzner. Finding subtle motifs by branching from sample strings. *Bioinformatics*, 1(1):1–7, 2003.

[22] B. Raphael, L.T. Liu, and G. Varghese. A uniform projection method for motif discovery in DNA sequences. *IEEE Transactions on Computational biology and Bioinformatics*, 1(2):91–94, 2004.

[23] C. K. Reddy and H.D. Chiang. A stability boundary based method for finding saddle points on potential energy surfaces. *Journal of Computational Biology*, 13(3):745–766, 2006.

[24] M. Sagot. Spelling approximate or repeated motifs using a suffix tree. *Lecture Notes in Computer Science*, 1380:111–127, 1998.

[25] E. Segal, Y. Barash, I. Simon, N. Friedman, and D. Koller. From promoter sequence to expression: a probabilistic framework. In *Proceedings of the sixth annual international conference on Computational biology*, pages 263 – 272, Washington, DC, USA, 2002.

[26] M. Tompa and N. Li et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nature Biotechnology*, 23(1):137 – 144, 2005.

[27] M. Waterman, R. Arratia, and E. Galas. Pattern recognition in several sequences: consensus and alignment. *Mathematical Biology*, 46:515–527, 1984.

[28] E. Xing, W. Wu, M.I. Jordan, and R. Karp. LOGOS: A modular bayesian model for de novo motif detection. *Journal of Bioinformatics and Computational Biology*, 2(1):127–154, 2004.