BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 18 2006, pages 2217–2223
doi:10.1093/bioinformatics/btl371

Sequence analysis

Finding motifs from all sequences with and without binding sites

Henry C. M. Leung* and Francis Y. L. Chin
Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong

Received on April 11, 2006; revised on June 12, 2006; accepted on July 4, 2006
Advance Access publication July 26, 2006
Associate Editor: Nikolaus Rajewsky

ABSTRACT

Motivation: Finding common patterns, motifs, from a set of promoter regions of coregulated genes is an important problem in molecular biology. Most existing motif-finding algorithms consider a set of sequences bound by the transcription factor as the only input. However, we can get better results by considering sequences that are not bound by the transcription factor as an additional input.

Results: First, instead of using the simple hyper-geometric analysis, we propose to calculate the likelihood based on a more precise probabilistic analysis which considers motif length, sequence length and number of binding sites as input parameters for testing whether motif is found. Second, we adopt an heuristic algorithm bases on our analysis to find motifs. For the simulated and real datasets, our algorithm ALSE compares favorably against common motif-finding programs such as SeedSearch and MEME in all cases and performs very well, especially when each input sequence contains more than one binding site.

Availability: ALSE is available for download at the homepage http://alse.cs.hku.hk

Contact: cmleung2@cs.hku.hk

1 INTRODUCTION

Gene expression is the process whereby a gene, coding region in the genome, is decoded to produce protein. This process has two main steps, transcription and translation. During the transcription process, one or more molecules called transcription factors will bind to several special regions, called binding sites, in the promoter regions of the genes. Then the gene will be decoded to form a chain called mRNA. During the translation process, the mRNA will be decoded to produce the correlated protein.

Understanding the regulatory mechanisms that control gene expression is an important problem in molecular biology. A subproblem is to locate the set of binding sites in the promoter regions of the genes. Although a transcription factor can bind to several-binding sites, these binding sites should have similar pattern and length. The motif-finding problem is to find the binding sites and the common patterns, motifs, of these sites from a set of sequences suspected to be bound by some transcription factors (strong-signal sequences).

Different models have been developed to solve this problem. Bulher and Tompa (2002), Chin and Leung (2006, 2005a, b), Leung and Chin (2005a), Li *et al*. (2002) and Pevzner and Sze (2000) use a string to represent a motif. They assume each strong-signal sequence contains at least one substring that is similar to the motif in terms of Hamming distance or number of substitutions. Bailey and Elkan (1994, 1995), Leung and Chin (2005b), Eskin (2004), Hughes *et al*. (2000), Lawrence *et al*. (1993) and Liu *et al*. (1995) assume the strong-signal sequences are constructed according to a background occurrence probability of the nucleotides in non-binding regions with implanted substrings generated according to a probability matrix that represents the motif. All these models have a common weakness that only strong-signal sequences are used as input. As these models cannot confirm whether those discovered patterns also occur in other sequences which are not supposed to contain any binding sites, these models fail to find the correct motifs.

Helden *et al*. (1998), Jensen and Knudsen (2000), Sinha (2002) and Leung and Chin (2005c) treat those sequences that are ‘not’ bound by the transcription factor (weak-signal sequences) as additional input. Since the weak-signal sequences are not bound by the transcription factor, they should not contain many patterns similar to those binding sites. Motifs are those patterns that occur frequently in strong-signal sequences but rarely in weak-signal sequences. These algorithms discover motif candidates from the strong-signal sequences and perform a post-process to filter out those candidates exist frequently in the weak-signal sequences. Barash *et al*. (2001) and Segal *et al*. (2002) discovered the motifs directly from the strong-signal and weak-signal sequences. Barash *et al*. (2001), based on hyper-geometric analysis on two sets of input sequences, derive the probability that the set with strong-signal sequences has proportionally more sequences with binding sites than the other set with weak-signal sequences. This probability is used as the ‘testing function’ whether the discovered motif is correct. When this probability is small, it is plausible that this difference in the numbers of sequences with binding sites in these two sets is not an artifact and hidden motif can be found. Segal *et al*. (2002) use a Bayesian network to model the relationship among the DNA sequences (including both strong-signal and weak-signal sequences), transcription factors and gene expression levels, and to find a motif which can best-fit the experimental results. However, both Barash *et al*. (2001) and Segal *et al*. (2002) assume each sequence has at most one binding site. Without considering the fact that a motif can occur in a sequence more than once, it may fail to find the hidden motif even when the number of binding sites is sufficiently large but the number of strong-signal sequences containing the motif is small. Note that even if each sequence contains at most one binding site, the testing function based on hyper-geometric analysis has made some assumptions that may affect the accuracy of the algorithm, e.g. the length of the motif, the length of the sequences and the number of binding sites are not considered in their testing function.

In this article, we assume that a motif is represented by a probability matrix and each sequence can have more than one binding

*To whom correspondence should be addressed.

site. Given a predicted motif matrix, we calculate the likelihood of this matrix being the hidden motif. If this likelihood is large, the predicted motif should be biologically significant with respect to the input sequences. Since our new testing function is more precise and can handle sequences with more than one binding site, our algorithm ALSE (stands for ALl SEquences) based on the new model and testing function outperforms the popular algorithm MEME (Bailey and Elkan, 1994) which uses only strong-signal sequences as input when a set of weak-signal sequences is available. It also works better than SeedSearch (Barash *et al*., 2001) in finding motif in sequences especially those with multiple binding sites.

This article is organized as follows. We describe the shortcomings of the hyper-geometric analysis in Section 2. In Section 3, we explain our model and how to estimate the likelihood of a matrix being the hidden motif. An algorithm ALSE to find the motif is introduced in Section 4. Experimental results of the algorithm on real data and simulated data are described in Section 5, followed by a discussion in Section 6.

2 HYPER-GEOMETRIC MODEL AND ITS SHORT-COMING

2.1 Hyper-geometric model

In Barash *et al*. (2001), a 4 × w probability matrix M is used to represent a motif of length w, where $M(c, k)$ represents the occurrence probability of nucleotide c in the k-th position of a binding site. The log likelihood of a length-w string σ generated from a probability matrix M is

$$\text{score}(\sigma, M) = \log \prod_{i=1}^{w} M(\sigma[i], i) \tag{1}$$

where $\sigma[i]$ is the i-th nucleotide of sequence σ. A sequence is predicted as containing binding site of motif matrix M if it has a substring σ with score(σ, M) larger than some threshold α.

Barash *et al*. (2001) use the hyper-geometric analysis to determine whether a matrix M is likely to be the hidden matrix with the assumption that each sequence contains at most one binding site. With a given matrix M and threshold α, assume that there exist t sequences with binding sites in set T and f sequences with binding sites in set F. They then consider the scenario of setting a threshold for a random matrix such that there are in total $t + f$ sequences with binding sites in $|T| + |F|$ random sequences altogether. Under this scenario, they further calculate the probability that t or more sequences with binding sites are in set T.

$$p\text{-value} = \sum_{i=t}^{t+f} \frac{\binom{T}{i}\binom{F}{t+f-i}}{\binom{T+F}{t+f}} \tag{2}$$

If this probability (p-value) is small, matrix M is likely to be the hidden matrix for the sets of sequences, T and F. Thus, they want to find a matrix M and a threshold α to minimize the p-value.

2.2 Shortcomings of hyper-geometric model

Although this is a simple and reasonable way to test whether a hidden matrix is found, this approach has three shortcomings, whereas the first is about multiple binding sites while the other two are about the appropriateness of hyper-geometric analysis even when each sequence contains at most one binding site.

First, the hyper-geometric analysis [Equation (2)] might not be easily extensible to deal with sequences with multiple binding sites. One might intuitively model the probability of multiple binding sites in a sequence by the probability of placing $t + f$ balls (representing the total number of binding sites in the sequences) in $|T| + |F|$ urns with the assumption that each urn might take more than one ball. However, this model has over-simplified the situation by assuming that the probability of placing a ball into an urn is always the same, no matter whether the urn is empty or not. In the actual situation, the probability of a sequence taking up one more binding site is different when the sequence has different numbers of binding sites. This probability should be related to the length of the motif and the length of the sequences. Since the length of the motif and the length of the sequences are not considered in the hyper-geometric analysis, it is unlikely that it can be extended to deal with sequences with multiple binding sites easily.

Second, two matrices with the same p-value does not mean that these two motifs have the same probability to be the hidden motif. For example, assume T and F each containing 15 sequences, the probabilities (p-values) for the outcomes $(t, f) = (5, 5)$ and $(t, f) = (10,10)$ are the same according to the hyper-geometric analysis. However, the probabilities that there exists a threshold for a random matrix such that there are 10 or 20 sequences with binding sites are different. In fact, the probability of having 10 sequences with binding sites is smaller than that for 20 sequences. Thus the probability for the outcome $(t, f) = (5, 5)$ should be smaller and the corresponding matrix M found is more likely to be the hidden matrix.

Third, the hyper-geometric approach [Equation (2)] only considers the probability (p-value) that there are t or more sequences with binding sites in set T and the total number of sequences with binding sites is 'exactly' $t + f$. However other cases when the total number of sequences with binding sites is not $t + f$ are not considered. Equation (2) is only valid if we can always find the threshold such that there are exactly $t + f$ sequences with binding sites with respect to a random matrix. Since it is not always possible to find such a threshold, the probabilities of such cases, when the total number of sequences with binding sites is not $t + f$, have not been considered in the analysis.

In order to overcome these shortcomings, we introduce a new testing function which is the likelihood of the matrix M being the hidden motif. Instead of finding the matrix with the smallest p-value, we find the matrix with the largest likelihood.

3 SYSTEM AND METHODS

3.1 Calculating likelihood

In this section, we will describe how to calculate the (relative) likelihood $L(M \mid T \wedge F)$ of a matrix M being the hidden matrix. Similar to Bailey and Elkan (1994, 1995), Barash *et al*. (2001), Eskin (2004), Hughes *et al*. (2000), Leung and Chin (2005b), Lawrence *et al*. (1993) and Liu *et al*. (1995), we use a 4 × w probability matrix to represent the motif. Given a threshold α, a length-l substring σ in the input sequences T and F is predicted as a binding site of M if score(σ, M) $\geq \alpha$. We assume both the strong-signal sequences T and weak-signal sequences F are generated independently according to some background probability $B = \{p_A, p_C, p_G, p_T\}$, $p_A + p_C + p_G + p_T = 1$. Let α^* be the hidden threshold, we assume there is atleast one substring σ in T with score(σ, M^*) = α^*. Since sequences in T should contain relatively more binding sites of the hidden matrix M^* than sequences in F, we assume

$d_t > 0$ substrings with score(σ, M^*)$\geq\alpha^*$ are planted in random positions without overlapping in the sequences in T and $d_f > 0$ substrings with score(σ, M^*) $\geq \alpha^*$ are removed from the sequences in F.

Given t^* = the number of binding sites of M^* in T with respect to α^*, this means that there were $x_t = t^* - d_t$ binding sites of M^* in T before the d_t binding sites were planted. Given t binding sites in T, we approximate, through the conditional probability $P(T \mid M \wedge \alpha)$ that the set of sequences T are generated by the hidden matrix M with the hidden threshold α, the number of planted binding sites in T, as follows:

$$\begin{aligned}
P(T \mid M \wedge \alpha) &\propto P(t^* = t \mid M^* = M \wedge \alpha^* = \alpha) \\
&= \sum_{d'=1}^{t} P(t^* = t \wedge d_t = d' \mid M^* = M \wedge \alpha^* = \alpha) \\
&= \sum_{d'=1}^{t} P(x_t = t - d' \wedge d_t = d' \mid M^* = M \wedge \alpha^* = \alpha) \\
&= \sum_{d'=1}^{t} (P(x_t = t - d' \mid d_t = d' \wedge M^* = M \wedge \alpha^* = \alpha) \\
&\qquad P(d_t = d' \mid M^* = M \wedge \alpha^* = \alpha))
\end{aligned}$$

Assuming $P(d_t = d' \mid M^* = M \wedge \alpha^* = \alpha) = P(d_t = d')$ is uniformly distributed, we have

$$P(T \mid M \wedge \alpha) \propto \sum_{d'=0}^{t-1} P(x_t = d' \mid M^* = M \wedge \alpha^* = \alpha). \tag{3}$$

Similarity, let f^* be the number of binding sites of M^* in F with respect to α^*, i.e. there were $x_f = f^* + d_f$ binding sites of M^* in F before removing the $d_f > 0$ binding sites, we have

$$P(F \mid M \wedge \alpha) \propto \sum_{d''=f+1}^{m|F|} P(x_f = d'' \mid M^* = M \wedge \alpha^* = \alpha), \tag{4}$$

where $m = \lfloor n/w \rfloor$ is the maximum number of non-overlapping binding sites in a length-n sequence.

Assuming that the probability that a threshold α being picked as the hidden threshold is independent of the hidden matrix M and is uniformly distributed, we can calculate the (relative) likelihood of a matrix M being the hidden motif as follows.

$$\begin{aligned}
L(M \mid T \wedge F) &= P(t^* = t \wedge f^* = f \mid M^* = M) \\
&\propto \sum_{\alpha} P(t^* = t \wedge f^* = f \mid M^* = M \wedge \alpha^* = \alpha) \times \\
&\qquad P(\alpha^* = \alpha \mid M^* = M) \\
&\propto \sum_{\alpha} \sum_{\substack{0 \leq d' \leq t-1, \\ f+1 \leq d'' \leq m|F|}} (P(x_t = d' \mid M^* = M \wedge \alpha^* = \alpha) \times \\
&\qquad P(x_f = d'' \mid M^* = M \wedge \alpha^* = \alpha).
\end{aligned} \tag{5}$$

Therefore, the likelihood of a matrix being the hidden matrix increases with the number of binding sites in T and decreases with the number of binding sites in F. We will describe how to calculate $P(x_t = t - d' \mid M^* = M \wedge \alpha^* = \alpha)$ and $P(x_f = f + d'' \mid M^* = M \wedge \alpha^* = \alpha)$ in the following section.

3.2 Probability of having x_t binding sites

Although there are infinite number of thresholds, since there are only 4^w length-w strings, there are at most 4^w possible thresholds α^* that satisfy the requirement that there is at least one substring in T with score(σ, M^*) = α^*. We sort the 4^w threshold values in decreasing order and denote α_u as the u-th threshold. For a length-w string σ generated randomly with equal occurrence probability of each nucleotide, the probability that score(σ, M) $\geq \alpha_u$ is $p_u = u/4^w$.

Given a random length-n sequence s, we calculate the probability $P_b(u, v, \{k_i\})$ that there are exactly b non-overlapping length-w binding sites at $s[k_i - w + 1 \cdots k_i]$, $i = 1, \cdots, b$ (k_i is the ending position of the i-th binding site in s) with respect to matrix M and any threshold α, $\alpha_u \leq \alpha \leq \alpha_v$ where $v \leq u$.

Consider a length-w string σ, it is a binding site for any threshold α in [α_u, α_v] if score(σ, M_r) $\geq \alpha_v$ and it is not a binding site if score(σ, M_r) $< \alpha_u$. Assume the probabilities of the score of each substring at different positions in s are all independent and the probability of a substring is not a binding site is $1 - p_u$. Note that the following calculation is only an approximation because the scores of two overlapping substrings are not independent. Depending on the position of the last binding site $s[k_b - w + 1 \cdots k_b]$, the probability $P_b(u, v, \{k_i\})$ of a random sequence s containing exactly b non-overlapping binding sites at positions $\{k_1, \cdots, k_b\}$ with respect to any threshold α in [α_u, α_v] can be calculated as follows.

Case I: $k_b > n - w$, when the position of the last binding site is close to the end of the sequence, it is impossible to have a binding site after k_b.

$$\begin{aligned}
P_b(u, v, \{k_i\}) &= (1 - p_u)^{k_1 - w} p_v \cdots (1 - p_u)^{k_b - k_{b-1} - w} p_v \\
&= (1 - p_u)^{k_b - bw} p_{v.}^b
\end{aligned}$$

Case II: $k_b \leq n - w$ when the substring $s[k_b + 1 \cdots n]$ is longer than the motif length w

$$\begin{aligned}
P_b(u, v, \{k_i\}) &= (1 - p_u)^{k_1 - w} p_v \cdots (1 - p_u)^{k_b - k_{b-1} - w} p_v \\
&\quad \cdot (1 - p_u)^{n - k_b - w + 1} \\
&= (1 - p_u)^{n - (b+1)w + 1} p_v^b.
\end{aligned}$$

Note that this probability does not depend on the positions of the binding sites except the ending position of the last binding site k_b.

$$P_b(u, v, \{k_i\}) = \begin{cases} (1 - p_u)^{k_b - bw} p_v^b & \text{if } k_b > n - w \\ (1 - p_u)^{n - (b+1)w + 1} p_v^b & \text{if } k_b \leq n - w. \end{cases}$$

By considering all the possible positions for the binding sites, we can calculate the probability $P_B(u, v, b)$ that a length-n sequence contains exactly b binding sites with respect to a particular matrix M and any threshold α in [α_u, α_v].

$$\begin{aligned}
P_B(u, v, b) &= \sum_{\text{all possible } \{k_i\}} P_b(u, v, \{k_i\}) \\
&= \sum_{k_b = bw}^{n - w - 1} \left[\binom{k_b - bw + b - 1}{b - 1} (1 - p_u)^{n - (b+1)w + 1} p_v^b \right] \\
&\quad + \sum_{k_b = n - w}^{n} \left[\binom{k_b - bw + b - 1}{b - 1} (1 - p_u)^{k_b - bw} p_v^b \right].
\end{aligned}$$

Given X random length-n sequences, the following equation gives the probability $P_{X,x}(u, v)$ that there are exactly x binding sites in the X sequences with respect to a particular matrix M and any threshold α in [α_u, α_v].

$$P_{X,x}(u, v) = \sum_{\substack{\text{all possible } \{a_b\} \text{ s.t.} \\ \sum a_b = X \text{ and } \sum b a_b = x}} \frac{X!}{\prod_b a_b!} \cdot \prod_b P_B(u, v, b)^{a_b},$$

where a_b is the number of sequences with exactly b binding sites.

The probability that, using α_u as the threshold, T has exactly x_t binding sites of matrix M before planting any binding site is $P_{|T|, x_t}(u, u) - P_{|T|, x_t}(u, u - 1)$. We have to minus $P_{|T|, x_t}(u, u - 1)$ because we assume there is at least one substring σ in T with score (σ, M) = α_u. Similarly, the probability that, using α_u as thethreshold, F has exactly x_f binding sites of matrix M before removing any binding site is

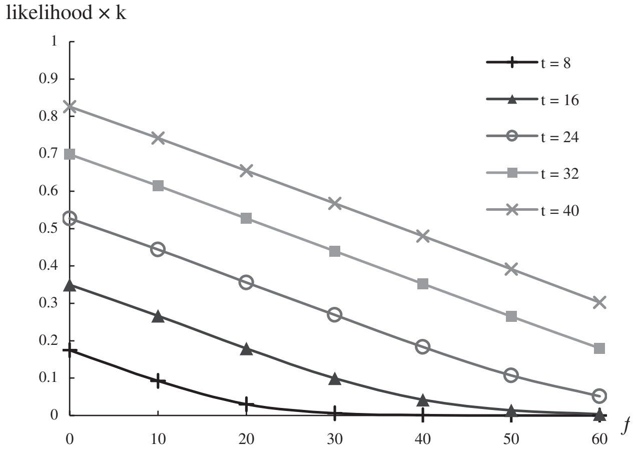

Fig. 1. Relative values of $L(M \mid T \wedge F)$ when $w = 7$ and $n = 250$. k is a constant for normalization.

$P_{|F|,x_f}(u,u)$. Substituting these two probabilities in Equation (5), we have

$$L(M|T \wedge F) \propto \sum_{u=1}^{4^w} \sum_{\substack{0 \le d' \le t-1, \\ f+1 \le d'' \le m|f|}} ((P_{|T|,d\prime;}(u,u) - P_{|T|,d\prime}(u,u-1)) \; P_{|F|,d''}(u,u)). \tag{6}$$

Thus the matrix M with the largest likelihood $L(M \mid T \wedge F)$ is likely to be the hidden motif.

3.3 Interpretation of the testing function

Figures 1–3 show the values of $L(M \mid T \wedge F)$ calculated using Equation (6) for different sequence length n when $w = 7$, $|T| = 20$ and $|F| = 50$. As shown in the figures, the likelihood of M being the hidden matrix increases with t, the number of binding sites in T, and decreases with f, the number of binidng sites in F. Moreover, when the sequence length n increases, the number of length-w* substrings in the input sequences increases, so as the expected number of binding sites of M*. Therefore, for a fixed t, i.e. the number of binding sites in T, the likelihood $L(M \mid T \wedge F)$ decreases with the increment of the sequence length n. In the extreme case when n tends to infinity, the number of binding sites of M* in T tends to infinity and the likelihood of a matrix with fixed number of binding sites approaches zero.

4 ALGORITHM

We use an heuristic algorithm called ALSE (stands for ALl SEquences) to find the motif based on our model. ALSE contains two main parts. The first part is to find a set of probability matrices M with large $L(M \mid T \wedge F)$ as seeds. The second part is the heuristic iterative step to refine these seed matrices to search for the hidden motif. We will first discuss the second part, the refining steps in Section 4.1 and then the method of finding seeds in Section 4.2.

4.1 Refining candidate matrix

Given a probability matrix M, an heuristic iterative procedure is applied to refine it to another matrix M' with larger $L(M \mid T \wedge F)$. Algorithm ALSE first calculates the probability of each substring in set T being a binding site. Based on these probabilities, matrix M is refined to M' such that those binding sites according to M will yield a higher score with respect to M'. Although this refinement increases the value of $L(M \mid T \wedge F)$ in practice, there is no guarantee that $L(M \mid T \wedge F)$ will increase in each step because of the effect of the sequences in F. By applying a similar approach as simulated annealing, we perform the above refinement even $L(M \mid T \wedge F)$ may decrease. However, after the first few refinements (five refinements in our experiments), M is refined to M' if and only if M' has a larger $L(M \mid T \wedge F)$, otherwise, the refinement will stop.

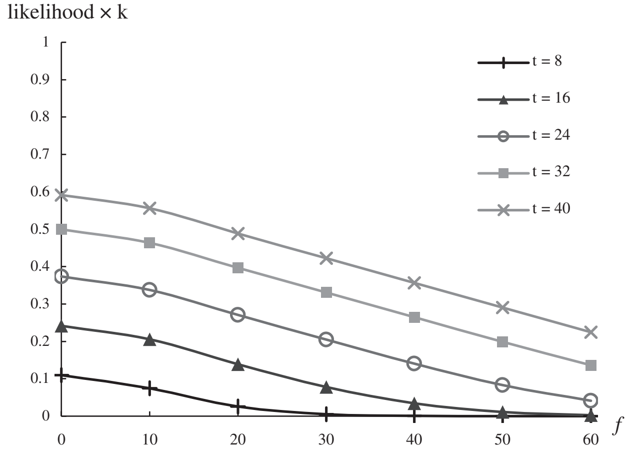

Fig. 2. Relative values of $L(M \mid T \wedge F)$ when $w = 7$ and $n = 500$. k is a constant for normalization.

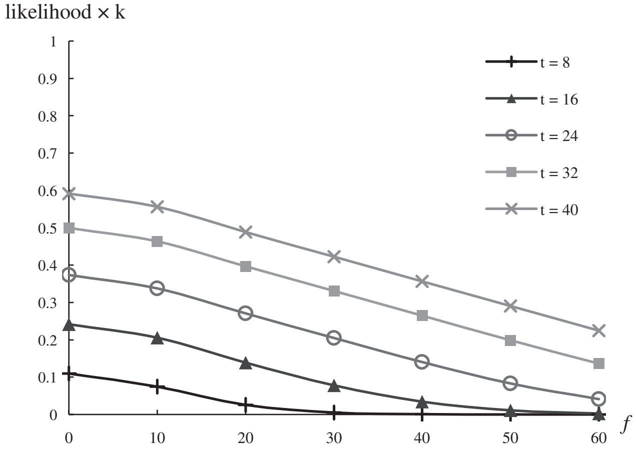

Fig. 3. Relative values of $L(M \mid T \wedge F)$ when $w = 7$ and $n = 750$. k is a constant for normalization.

Algorithm ALSE can be described as follows:

Step 1: For each length-w substring σ in T, calculate the probability score(σ, M) that the substring σ is a binding site of matrix M.

Step 2: Align these substrings and calculate a refined matrix M' as follows

$$M'(c,k) = \frac{\sum_{\sigma \mid \sigma[k]=c} \text{score}(\sigma, M)}{\sum_{\sigma} \text{score}(\sigma, M)} \tag{7}$$

These two steps will be repeated until $L(M' \mid T \wedge F) \le L(M \mid T \wedge F)$ or the number of iterations reaches a predetermined value.

4.2 Finding seeds

When the motif is short, the 4^w seed matrices are constructed by a similar method as Bailey and Elkan (1994). Each seed matrix is converted from a length-w string σ such that for each column i, the value of $M(\sigma[i], i)$ is 0.5 and the rest are 0.5/3. This method works well when the motif is short. However, when the motif length w increases, the number of seed matrices increases exponentially and the running time of ALSE becomes unacceptably long. Therefore, we apply the Voting algorithm (Chin and Leung, 2005b; Leung and Chin, 2005a) to find a subset of length-w string motifs, convert them into seed matrices as discussed before and refine them (as discribed in Section 4.1) to obtain probability matrices with larger $L(M \mid T \wedge F)$.

5 RESULTS

We implemented Algorithm ALSE in C++. All experiments were run on a machine with 2.4 GHz CPU and 1 GB memory. Each experiment took reasonable time (normally within a few minutes) to find the motif.

5.1 Simulated data

We constructed two sets of length-n random sequences with background probability $B = \{0.25, 0.25, 0.25, 0.25\}$ of sizes 20 for sets T. A 4×7 probability matrix M was constructed by assigning each entry a random number in the range (0,1] under the uniform distribution and normalizing the sum of entries in each column to 1. A total of t binding sites generated according to M were planted in sets T. We generated 50 sequences for sets F similarly. However, we ensured that F contains less binding sites than the expected (calculated based on threshold α^* which is the the lowest score of the t binding sites planted in T). We compared the performance of MEME (a popular motif finding algorithm based on strong-signal sequences using EM algorithm), SeedSearch (algorithm based on hyper-geometric approach) and ALSE. Sequences in sets T and F were then taken as input for ALSE and SeedSearch. Since MEME takes strong-signal sequences as the only input, the 20 sequences in set T were taken as input for MEME. We use the default parameters for these algorithms except the length of motifs.

We have experimented with different values of t, i.e. different numbers of binding sites in set T and binding sites of different information contents. Information content of a binding site σ is defined as

$$\mathrm{IC}(\sigma) = \log \frac{\Pr(\sigma \text{ generated according to the motif matrix M})}{\Pr(\sigma \text{ generated according to the background model})},$$

which represents the amount of information of the motif contained in each binding site. The higher the information content, the easier for an algorithm to find the motif. We controlled the IC of the generated matrices by generating a new matrix repeatedly until the IC of the matrix is within our specified range. For each set of parameters, we repeated the experiments 50 times with different matrices and used MEME, SeedSearch and ALSE to predict the hidden motifs.

We determined whether a predicted motif is correct by calculating the performance ratio = $|$predicted binding sites $\cap$ planted binding sites$|$ / $|$predicted binding sites $\cup$ planted binding sites$|$. A binding site is predicted correctly if it overlaps with some planted binding sites. A high performance ratio indicates the predicted motif can discover the planted binding sites accurately. A predicted motif is correct if the performance ratio of it is at least 50%.

Instead of finding one motif, motif discovering algorithms usually output a list of predicted motifs. Besides, the hidden motif seldom occurs at the top of the predicted list especially when the amount of information in the input sequences is small (i.e. number of binding sites is small and the information content of each binding site is small). Therefore, instead of considering one predicted motif only, we scanned through the top 20 predicted motifs of each algorithm. An algorithm is said to predict the hidden motif correctly if the performance ratio of one of the 20 predicted motifs is at least 50%.

Table 1. Results of ALSE, SeedSearch and MEME on simulated data

	t	IC	Successful rate ALSE (%)	SeedSearch (%)	MEME (%)
1	20	7.21	72	52	48
2		7.70	80	64	58
3		8.61	92	78	74
4	30	7.21	86	46	56
5		7.70	88	74	72
6		8.61	96	94	82
7	40	7.21	100	62	72
8		7.70	100	86	96
9		8.61	100	100	100

There were 20 length-500 sequences in set T and 50 length-500 sequences in set F. t is the number of binding sites in set T. For each set of parameters, we repeated the experiments 50 times. IC is was the average information content of a binding site.

The results are shown in Table 1. Each entry represents the percentage (out of 50 experiments) of hidden motifs found by the corresponding algorithm. As shown in Table 1, the performance of ALSE was no worse than SeedSearch and MEME in all cases. In particular, when the number of binding sites in set T was large ($t \gg |T|$) and the information content of each binding site was small ($IC \leq 7.70$), i.e. rows 7 and 8 of Table 1, ALSE and MEME took advantage of the large number of binding sites in T to find the hidden motif. However, SeedSearch did not perform as good because of its assumption that each sequence contained at most one binding site. Note that although SeedSearch made an invalid assumption, its performance increased with the number of binding sites because it had a higher chance to start from a seed matrix close to the hidden matrix when the number of binding sites was large. When the number of binding sites in T was small, MEME had difficulties to find the motif because of noise (i.e. a randomly picked matrix has a high probability to get a similar likelihood as the hidden matrix). However, by considering the weak-signal sequences in F, ALSE and SeedSearch performed better than MEME as shown in rows 1, 2 and 3 of Table 1.

5.2 Real biological data

In the database SCPD (http://rulai.cshl.edu/SCPD/), coregulated genes with known binding sites of yeast are reported. The TRANSFAC (http://www.gene-regulation.com/) contains similar information of other species. We tested the performance of the three algorithms MEME, SeedSearch and ALSE on these real biological data. We have performed experiments on those transcription factors with at least two published binding sites and the length of motif are at most eight only. For each set of coregulated genes, we took the 450 bp upstream and 50 bp downstream of the transcription start site (TSS) as the strong-signal sequences T. We randomly picked the 450 bp upstream and 50 bp downstream of the transcription start site of other genes as the weak-signal sequences. Similar to the experiments on simulated data, we evaluated the performance of these

Table 2. Results of ALSE, SeedSearch and MEME on the set of data in TRANSFAC

Factor name	Performance ratio		
	ALSE (%)	SeedSearch (%)	MEME (%)
ANTP	33.33	25	25
AS-CT3	100	100	33.33
BAS1	75	0.00	0.00
BEAF-32B	33.33	28.57	0.00
BEF-1_7	75	100	66.67
Bfactor	66.67	22.22	22.22
Cad	50	0.00	0.00
Cut	0.00	100	0.00
Da	100	100	0.00
DTF-1	50	0.00	25
E4b	100	38.46	0.00
EcR_7	50	33.33	33.33
En_7	11.11	8.33	10
FTZ-F1	50	100	100
GATA	100	0.00	0.00
GCN4	100	65	0.00
HSTF_5	80	50	16.67
RAP1	80	75	50
TAB	25	33.33	33.33
T-Ag	6.06	3.33	7.69
Ttk88K	100	100	0.00
Ubx_a_7	100	50	0.00
v-myb_7	33.33	33.33	0.00
	ALSE	SeedSearch	MEME
Number of times getting the best performance among three algorithms	19	8	3

three algorithms by calculating the performance ratio $= |\text{predicted binding sites} \cap \text{published binding sites}| / |\text{predicted binding sites} \cup \text{published binding sites}|$. A high performance ratio indicates the algorithm can predict the binding sites accurately. Table 2 shows the the performance of these three algorithms on all the data of the yeast and fruitfly. (Results are not shown when all three algorithms had similar performance.)

In these 23 datasets ($\sim$30% of the data), ALSE had the best performance among three algorithms 19 times. It is because ALSE can handle multiple binding sites and utilize the additional information about the motif from the weak-signal sequences in F. When compared with the performance of SeedSearch and MEME, we can conclude that the weak-signal sequences contain much information about the motif. Even for SeedSearch which does not consider multiple binding sites in a sequence, it still can outperform MEME as indicated in Table 2.

6 DISCUSSION

In this article, we have introduced a new model for motif finding. By considering those sequences that are not bound by a transcription factor, we can eliminate those patterns that appear frequently in every part of the genome as motifs. SeedSearch is based on this idea for finding motif. Unfortunately SeedSearch assumes that each sequence has at most one binding site and uses a simplified testing function in its iterative step. Our proposed algorithm uses a more accurate testing function (which considers the length of the motif and also the length of the input sequences) and can handle sequences with multiple binding sites. For the simulated and real datasets, ALSE performs favorably when compared with the common motif-finding program MEME and SeedSearch.

Like other algorithms, ALSE cannot guarantee that the optimal $M*$ with the largest $L(M\,|\,T \wedge F)$ will be found and the success of the algorithm depends on how the seed matrices are selected. In our algorithm, a limited number of promising seeds are selected from the 4^w matrices generated from all the possible 4^w length-w strings. However, this approach offinding seeds will fail when the length of the motif is large. Our future direction is to consider some new seed-finding methods with acceptable time complexity and effectiveness.

ACKNOWLEDGEMENTS

The authors are very thankful to the reviewers for their useful comments. This research was supported in parts by the Hong Kong RGC grant HKU 7135/04E.

Conflict of Interest: none declared.

REFERENCES

Bailey,T. and Elkan,C. (1994) Fitting a mixture model by expectation maximization to discover motifs in biopolymers. In *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology,* **2**, 28–36.

Bailey,T. and Elkan,C. (1995) Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Mach. Learn. J.*, **21**, 51–83.

Barash,Y. *et al.* (2001) A simple hyper-geometric approach for discovering putative transcription factor binding sites. *Proc. WABI*, **1**, 278–293.

Bulher,J. and Tompa,M. (2002) Finding motifs using random projections. *J. Comput. Biol.*, **9**, 225–242.

Chin,F. and Leung,H. (2006) An efficient algorithm for string motif discovery. *Proc. APBC*, **4**, 79–88.

Chin,F. and Leung,H. (2005a) An efficient algorithm for the extended (l,d)-motif problem with unknown number of binding sites. *Proc. BIBE*, **5**, 11–18.

Chin,F. and Leung,H. (2005b) Voting algorithm for discovering long motifs. *Proc. APBC*, **3**, 261–271.

Eskin,E. (2004) From profiles to patterns and back again: a branch and bound algorithm for finding near optimal motif profiles. *Proc. RECOMB*, **8**, 115–124.

Helden,J. *et al.* (1998) Extracting regulatory sites from the upstream region of yeast genes by computational analysis of oligonucleotide frequencies. *J. Mol. Biol.*, **281**, 827–842.

Hughes,J.D. *et al.* (2000) Computational identification of *cis*-regulatory elements associated with groups. *J. Mol. Biol.*, **296**, 1205–1214.

Jensen,L.J. and Knudsen,S. (2000) Automatic discovery of regulatory patterns in promoter regions based on whole cell expression data and functional annotation. *Bioinformatics*, **16**, 326–333.

Lawrence,C. *et al.* (1993) Detecting subtle sequence signals: a Gibbs sampling strategy. *Science*, **262**, 208–214.

Leung,H. and Chin,F. (2005a) Algorithms for challenging motif problems. *JBCB*, **4**, 43–58.

Leung,H. and Chin,F. (2005b) Finding exact optimal motif in matrix representation by partitioning. *Bioinformatics*, **21**, ii86–ii92.

Leung,H. and Chin,F. (2005c) Generalized planted (*l*,*d*)-motif problem with negative set. *WABI*, **5**, 264–275.

Li,M. *et al.* (2002) Finding similar regions in many sequences. *J. Comp. Syst. Sci.*, **65**, 73–96.

Liu,J.S. *et al.* (1995) Bayesian motifs for multiple local sequence alignment and Gibbs sampling strategies. *J. Am. Stat. Assoc.*, **90**, 1156–1170.

Pevzner,P.A. and Sze,S.H. (2000) Combinatorial approaches to finding subtle signals in DNA sequences. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **8**, 269–278.

Segal,E. *et al.* (2002) From promoter sequence to expression: a probabilistic framework. *Proc. RECOMB*, **6**, 263–272.

Sinha,S. (2002) Discriminative motifs. *Proc. RECOMB*, **6**, 291–298.

APPENDIX

In this section, we give detail description of the experiments.

When performing experiments on MEME, we use the command 'meme [input sequences file path] -dna -w [motif length] -nmotifs 20 [output file path]' for the simulated data and the command 'meme [input sequences file path]-dna -revcomp -w [motif length] -nmotifs 20 [output file path]' for the real biological data. When performing experiments on SeedSearch, we use the command 'seed-n 20 -l [motif length] [input sequences file path] [wighted file path] [output file path]' for the simulated data. weighted file is a file indicates which sequences are in T and which sequences are in F. And we use the command 'seed -n 20 -l [motif length] -reverse [input sequences file path] [wighted file path] [output file path]' for the real biological data.