

Discovering Motifs from Biosequences Based on Instance Density

Ya Zhang* Chao-Hisen Chu* Hongyuan Zha† Yixin Chen‡ Xiang Ji§

*School of Information Sciences and Technology, The Pennsylvania State University

†Department of Computer Science and Engineering, The Pennsylvania State University

‡Department of Computer Science, University of New Orleans

§NEC Laboratories America, Cupertino, CA

Abstract

The importance of motifs in biological systems is unquestioned. With the increasing accessibility of sequential information such as DNA and protein sequences, in-silico motif discovery has been an important task of bioinformatics. This paper presents a motif discovery algorithm based on instance density, where a motif is seen as a unique feature that distinguishes the set of sequences with the motif from other sequences. The algorithm implicitly takes into account both the background information and the motif instance information. We applied the algorithm to discovering transcriptional factor binding sites in yeast promoter regions. The results showed that the proposed algorithm outperformed a benchmarked method. In 15 out of 22 cases, the highest-scoring motifs reported by our algorithm closely matched the biologically-mapped motifs.

1 Introduction

Discovering motifs in biosequences has been a topic of immense study in recent years. Biological motifs are not only consensus patterns embedded in biosequences but also patterns with biological meaning. That is, they play important roles in the biological systems. In the case of protein sequences, short conserved patterns are generally closely related to protein functions and structures. For example, an enzyme catalytic site of a protein usually has a certain conserved pattern. DNA motifs, in the form of conserved short oligonucleotides, are often binding sites for gene regulators such as the transcriptional factors. Each transcriptional factor may have its own characteristic binding site. Hence, discovering DNA motifs plays an important role in understanding transcription control, which is one of the most essential levels of gene regulation. To date there are over two hundred transcriptional factors with characteristic binding sites verified in Yeast, but many more need to be identified. Since different algorithms may fit different applications, in

this paper, we mainly focus on the discovery of DNA motifs as transcriptional factor binding sites.

In its simplest form, the problem of discovering DNA motifs can be generically formulated as follows.

Given a set of DNA sequences, each of which is known to bind to the same transcriptional factor¹, find out the consensus pattern of the unknown binding sites.

There are two main challenges faced in this field of research. First, because the motifs are subject to various kinds of mutation such as substitution, deletion and insertion, their occurrences in DNA sequences are not necessarily the same. As a result, it is difficult to find a proper representation for the motifs. Second, due to random occurrences of some oligonucleotides with patterns similar to the motifs in the background sequences, it is usually impossible to recover the exact location of motif occurrences in the DNA sequences[4]. The development of automated computational methods for motif finding in biosequence has thus proven difficult. Despite of many previous efforts, this problem is far from being resolved.

In this paper, we introduce a novel motif mining algorithm based on instance density in the motif space. We seek to maximize the probability that the motif is shared among the set of training sequences. This probability measures the fitness of the motif as the feature of the training sequences. In the mean time, we also intend to minimize the probability of the motifs being generated from background model. The latter probability represents the uniqueness of the motif as the feature of the training sequences. The instance density of a motif in a set of sequences is then defined as the ratio of these two probabilities. The notions behind the algorithm are:

1. The true motifs (i.e. the experimentally-mapped binding sites of transcriptional factors) are more likely to occur in each DNA sequence at least once because the presents of the motifs is necessary for the binding of transcriptional factors to promoters. The motifs with

at least one occurrences in each of the training promoter sequences are more likely to be the true motifs than those do not.

2. The true motif should be distinguishable from the background pattern because the transcriptional factors should not binding to any random sequences. Thus, motifs have higher possibility of being the true motifs if they are less likely being generated by the background model.

The algorithm is developed and applied to 22 yeast promoters data sets, each containing promoter sequences binding to a transcriptional factor, to find the common transcriptional factor binding sites. The experimental results, compared with those of Dmotif algorithm[9], have demonstrated the robustness of our algorithm in mining the binding sites.

2 Previous Work

In the past several years, many motif discovery algorithms have been proposed, including greedy algorithms[3], Expectation Maximization(EM) algorithms[1], Gibbs Sampling[6, 7, 10], and other algorithms. Depending on the way of defining the motif search space, these methods generally fall into two categories: pattern-driven algorithms[8, 9] and sample-driven algorithms[2, 4, 6, 7, 10, 11]. Assuming a motif of length l is over alphabet set A , pattern-driven algorithms treat all $|A|^l$ l -letter patterns as a candidate motif, where $|A|$ is the size of the alphabet. Some criteria is used to rank the candidates and select the most likely candidates. Pattern-driven algorithms suffer from the curse of dimensionality in the sense that the computational cost becomes inhibitive when l is very large. For example, when k is 4 and l is 10, there are over one million candidates for a motif. An alternative approach is to limit the motif search space to the patterns that actually appear in the sample sequences, as sample-driven algorithms do. However, most sample-driven algorithms involve an optimization step. Thus, the performance of these algorithms rely heavily on the starting points. They may terminate in a local optimum motif rather than the biologically functional motif, due to the lack of a proper starting point.

Buhler and Tompa[2] proposed a projection-based approach ‘PROJECTION’ to pick up starting points for EM algorithms. Their PROJECTION algorithm achieved good performance while lowering the computational cost by reducing the number of starting points for refinement. The algorithm is also applicable to optimization strategies other than EM algorithms.

Another way to avoid the problem of ending in local optimum is to construct a hybrid approach from pattern-driven

and sample-driven approaches. Keich and Pavzner[4] designed a MULTIPROFILER algorithm, where a neighborhood of each segment in the DNA sequences is used as a possible motif. They utilized multi-positional profiles for motif discovery and showed that the algorithm can find subtle motifs, whose scores, according to[4], are “unremarkable compared with those of some random motifs presented in the same sample”.

Sinha[9] proposed the Dmotif algorithm to address the problem from a feature selection perspective. Each candidate motif is viewed as a feature. Classifier are built on each of these features to discriminate input promoter regions from a set of randomly-generated background promoters. The ones with smallest classification errors are then reported as the likely patterns.

Despite of these efforts, the problem of discovering motif is far from being resolved because of the complexity inherent to problem. The Dmotif algorithm, one of the state-of-the-art motif discovery algorithms, has failed to identify known transcriptional factor binding sites in several cases. In this study, we intend to bridge the gaps by introducing an instance density-based approach.

3 Methods and Implementation

3.1 Background Modelling

The DNA sequences are usually much longer than the unknown patterns, which is one of the main sources of noise. As a result, one of the most important steps in motif discovery is to model background sequences. Many probabilistic-based motif discovery methods represent the background sequences based on single nucleotide frequency in the data set[5, 6]. However, this background model requires a strong independence assumption. Therefore, it cannot fully capture the complex structure of DNA sequences[10]. Our explorations have demonstrated that this background modelling method is inappropriate. At many times the true patterns cannot be distinguished from the random ones with this type of background model.

In the literature, various context-dependent background models have been proposed using weaker independence assumptions. Most of the recent algorithms use a higher-order Markov chain model to represent DNA sequences. With a Markov chain model of order m , the probability of nucleotide i occurring at position j depends only on the m previous nucleotides in the sequence. Thus, the probability of a sequence being generated by background model, B_m , is given by

$$P(s|B_m) = P(s_1, \dots, s_m) \prod_{l=m+1}^L P(s_l|s_{l-1}, \dots, s_{l-m}), \quad (1)$$

where s is a DNA sequence of length L with nucleotide s_i in the i^{th} position, and B_m is the background model of order m . The probability $P(s_l|s_{l-1}, \dots, s_{l-m})$ and $P(s_1, \dots, s_m)$ can be estimated from a set of DNA sequences by counting all oligonucleotides of length m and $m+1$. This estimation usually gives a good approximation of the true biological model when the set of DNA sequences used for parameter estimation is large. Thijs *et al.*[11] compared various background models and showed that the use of a context-dependent model based on a higher-order (3^{rd} -order or 4^{th} -order) Markov process significantly enhanced the performance of the motif finding algorithm in the presence of noisy data. Therefore, our experiments employ the 3^{rd} -order Markov chain model:

$$P(s|B_3) = P(s_1, s_2, s_3) \prod_{l=4}^L P(s_l|s_{l-1}, s_{l-2}, s_{l-3}). \quad (2)$$

3.2 The Motif Finding Algorithm Based on Instance Density (MFA_{ID})

Representing motifs is the very first decision in designing a motif discovery algorithm. We represent motifs as consensus strings, which are strings over the International Union Of Pure And Applied Chemistry (IUPAC) degenerate symbols ($\{A, C, G, T, R, S, W, M, Y, K, N\}$) that are restricted expressions over $\{A, C, G, T\}$.

Suppose we have a set of DNA sequences $S = \{s_1, s_2, s_3, \dots, s_N\}$, where s_i ($i \in \{1, 2, \dots, N\}$) is a DNA sequence with length L . L is the same for all DNA sequences in the set. We also assume that a certain unknown motif occurs at different unknown positions of each DNA sequence. The motif is modelled as a consensus with mismatches. A candidate motif is a regular expression over the alphabet $\{A, C, G, T\}$. Due to the mismatches, the occurrences of the motif in the DNA sequences may differ significantly. However, they should all closely match the motif pattern. For example, *ATCAGC* and *ACCGGC* differ from each other in two out of six positions, but they both might be mutated from the same pattern *ATCGGC*. For a motif of length l ($l \ll L$), we define its instance in a DNA sequence s_i ($i \in \{1, 2, \dots, N\}$) as a oligonucleotide of length l in the sequence that best matches the motif. A motif or its mutation should have one or more occurrences in each sequence in the set S if the motif agrees with the set of sequences very well. By finding what is common among the set S , we can select some candidate motifs that may agree with the set S . However, the subtleties lie in:

1. The appearance of a motif in different sequences may vary, as a result of mutation;
2. there are chances that some oligonucleotides in the background sequences may closely match the motif,

which introduces considerable noises. Therefore, we should distinguish patterns that are over-represented by chance and motifs that have biological functions.

Here we assume that a motif would more likely have a biological function if the probability that it is generated from the background model by chance is low. Thus, we want to maximize the probability of a candidate motif c being the shared motif among the sequence set S while minimizing the probability of a candidate motif c being generated by the background model B_m with respect to c . This idea gives rise to the definition of instance density (ID) of a candidate motif c as:

$$ID(c) = \frac{P(c|S)}{P_{B_m}(c)}, \quad (3)$$

where $P(c|S)$ is the probability that the candidate c is the shared motif among the set S , and $P_{B_m}(c)$ is the probability that the candidate c is generated by the background model. The instance density of a motif is high if the motif is evenly distributed in different DNA sequences and if the motif is less likely to be generated by chance from the background model. Thus, the instance density of a motif measures the significance of a motif being the true motif for the set of DNA sequences. The problem now is to maximize the instance density with respect to c . Using Bayes Rule, (3) can be reformulated into

$$ID(c) = \frac{P(S|c)P(c)}{P_{B_m}(c)P(S)}. \quad (4)$$

where $P(S|c)$ is the probability of observing the set of sequences S , given the candidate motif c is the true motif. $P(S)$ is a constant with respect to c . In addition, since we do not have any prior knowledge about the distribution of c , we assume it be a uniform distribution, i.e. $P(c)$ is a constant. Therefore,

$$\begin{aligned} \arg \max_{c \in C} ID(c) &= \arg \max_{c \in C} \frac{P(c|S)}{P_{B_m}(c)} \\ &= \arg \max_{c \in C} \frac{P(S|c)}{P_{B_m}(c)}. \end{aligned} \quad (5)$$

As a result, the maximization problem is converted to maximize $\frac{P(S|c)}{P_{B_m}(c)}$ with respect to c .

In order to calculate $P(S|c)$, we introduce the notation of the distance between a motif and a set of DNA sequences. First, the distance between two oligonucleotides of length l is defined as the ratio of unmatched nucleotides when the two oligonucleotides of length l are aligned. The distance between the motif c and the DNA sequence s_i is then defined as the distance between the motif c and its instance in the DNA sequence s_i

$$dis(c, s_i) = \min_{s_{ij} \in s_i} dis(c, s_{ij}), \quad (6)$$

where s_{ij} is a segment of s_i with length l starting at the j^{th} position. This definition has a winner-takes-all flavor, in that, as long as the sequence S contains the motif, the winner will be the motif, and the distance will be small. The distance between a motif c and the set of DNA sequences S is then defined to be the average of all distances between the motif c and the individual DNA sequences

$$dis(c, S) = \frac{1}{N} \sum_{i=1}^N dis(c, s_i). \quad (7)$$

Intuitively, given a candidate motif c , the probability that the set of DNA sequences S agrees with the motif c is high if we can find a close match for c at each DNA sequence in the set. If a candidate motif is the true motif, then the distance between the candidate motif and the set of the DNA sequences is expected to be close to 0. Therefore, $P(S|c) \propto \exp^{-\alpha \times dis(c, S)}$. Thus, Eq. 5 can be reformulated as

$$\begin{aligned} arg \max_{c \in C} ID(h) &= arg \max_{c \in C} \frac{P(S|c)}{P_{B_m}(c)} \\ &= arg \max_{c \in C} \frac{\exp^{-\frac{\alpha}{N} \sum_{i=1}^N dis(c, s_i)}}{P_{B_m}(c)}. \end{aligned} \quad (8)$$

As mentioned before, some of the motifs may have small distances to the set of DNA sequences simply because of close matches by chance. By minimizing the probability of a candidate motif c being generated by the background model B_m with respect to c , we exclude those motifs that are more likely to be randomly generated from the background model. $P_{B_m}(c)$ can be computed by

$$P_{B_m}(c) = P(c_1, \dots, c_m) \prod_{i=m+1}^l P(c_i | c_{i-1}, \dots, c_{i-m}). \quad (9)$$

The overall flow of the algorithm can be summarized as follows.

Algorithm MFA_{ID}

Input: P , the set of promoter sequences known to bind to a transcriptional factor; l , motif consensus length; s , the number of spaces that allowed; and d , the number of degenerate symbols that allowed.

Output: Ten highest ranked putative motifs.

Build the motif search space C by an enumerative approach based on l , s and d .

For each candidate motif c in C do

Compute the distance between c and P according to Eq. 7.

If $dis(c, S)$ is less than a threshold t do

Compute the instance density of c ($ID(c)$) in P according to Eq. 3.

END

END

Sort the motifs in S according to non-decreasing order of $ID(c)$.

Report motifs with ranks less than 10.

4 Experiments and Results

4.1 Data sets

In order to build a Markov chain model for background sequences, we retrieve a data set with some of the yeast promoter sequences that are publicly available. With the promoter sequences retrieval tool provided at the Promoter Database of *Saccharomyces Cerevisiae* (SCPD)², 232 yeast promoters are retrieved and used to estimate the parameters for the Markov-chain-based background model.

SCPD catalogs more than 100 transcriptional factors. For each transcriptional factor, it provides information about genes under its regulation, experimentally-mapped binding sites at promoter regions of these genes, as well as a consensus binding site. To test the performance of our algorithm, test sets are built from SCPD for transcriptional factors occurring in no less than 3 promoters and having a consensus binding site. Totally 22 transcriptional factors are selected, and each data set contains the promoter sequence information of the corresponding genes and consensus binding site. The length of the promoter sequences are 600 bps.

4.2 Experiments

The proposed algorithm is used in mining the experimentally-mapped transcriptional factor binding sites in the 22 promoter sequence sets. A 3^{rd} -order Markov chain model is employed for modelling the background sequences. The parameters of the background model are estimated from the 232 yeast promoter sequences. The same background model is used all through the experiments.

Based on empirical analysis of the yeast promoter database, most of the motifs have a consensus length of 6 to 8. Therefore, it is feasible to use an enumerative approach to build the motif search space, which avoids the problem of terminating at local optimum. Similar to [9], a candidate motif in our experiment is a regular expression over the alphabet $\{A, C, G, T, R, Y, S, W, M, K, N\}$. According to previous findings [9, 12], if a motif contains spacer, the spacer usually presents at the middle of the motif with length from 1 to 11 base pairs. This feature is reflected in our candidate motif by allowing only 1 to 11 consecutive N's at the center of each motif.

Three key parameters are required by the algorithm: the length of consensus nucleotides, the length of spaces, and the number of degenerate symbols that are allowed. The candidate motifs were first tested for agreement with the set of sequences based solely on the distance between the candidate motif and the set of sequences. Only those motifs with distance less than a certain threshold were further

²<http://cgsigma.cshl.org/jian/index.html>

Table 1. Results of mining transcriptional factor binding sites in yeast promoter regions. The results of Dmotif algorithm[9] are included here for ease of comparison. * The reported motif overlaps with the binding sites cataloged in SCPD. Thus, it is considered a close match.

Regulon	Binding site	Our algorithm		Dmotif	
		Motif found	Rank	Motif found	Rank
ABF1	TCRNNNNNNACG	CACNNNNNNCGK	1	TCANNNNNNAMG	2
CPF1	TCACGTG	CACGTG	1	CACGTG	1
CSRE	YCGGAYRRRAWGG	CGGATGRA	1	CGGATGRA	8
SCB	CNCGAAA	GTSACGA	1	TCGCGAA	2
GAL4	CGGNNNNNNNNNNCCG	CGGNNNNNNNNNNCCG	1	CGGNNNNNNNNNNCCG	1
GCR1	CWTCC	CTTCC	46	CTTCC	13
HAP1*	CGGNNTANCGG	CGGMCGG	1	GGANNNNCGG	1
HSE	TTCNNGAA TTCNNGAA GAANNNTCC GAANNNTCC	TTCTAGA	1	TTMTAGAA	6
MCB	WCGCGW	MCGCGT	1	ACGCGT	1
MCM1*	CCNNNWRGG	TAGGAAA	1	TTTCCTAA	1
MATa2	CRTGTWWWW	CATKTWA	2	CATGTMA	2
MIG1	CCCCRNWWWWW	CCCCRG	12	MCCCCAG	1
PHO4	CACGTK	CACGTG	1	CACGTG	1
PDR3	TCCGYGGA	TCCGCGGA	1	TCCGYGGA	2
REB1	YYACCCG	YTACCCG	1	YTACCCG	1
ROX1	YYNATTGTTY	CCTATTG	1	CCTATTG	7
RAP1	RMACCCA	CCCARWC	1	ACCCAGW	1
CAR1	AGCCGCSA	TAGCYRC	17	TAGCCGCS	2
SFF	GTMAACAA	Not found	-	Not found	-
STE12	ATGAAA	ATGAAAC	1	ATGNAAC	1
TBP	TATAWAW	TATAWAW	13	Not found	-
UASPHR	CTTCCT	Not found	-	Not found	-

tested. The threshold was determined based on some empirical analysis on binding sites of some transcriptional factors.

4.3 Results

The computational results are summarized in Table 1. Column 1 lists the name of the transcriptional factors. Column 2 shows the biologically-mapped binding site consensus of the corresponding transcriptional factors. The first close match to the consensus binding site and its rank, reported by our algorithm, are presented in columns 3 and 4. The first close match reported by Dmotif algorithm and its rank are given in columns 5 and 6. It can be seen that in 15 out of 22 categories of promoters, the known consensus closely matches the top ranking motifs reported by our proposed algorithm. However, there are some cases that the algorithm failed to report true motifs in the first 10 highest-scoring motifs. In those cases, by scrutinizing the transcriptional factor binding sites in each promoter sequences, we found there are significant variations among sites presented in different positions of promoter regions.

Since the same set of data were used to test Sinha’s Dmotif algorithm[9], we compare the performance of our proposed algorithm with the published results of the Dmotif algorithm. We compute the percentage of cases in which

the first-ranked motif reported by the algorithms closely matches the known consensus. For our algorithm, this value is 68%, while the value for Dmotif algorithm is 45%. Thus, in terms of the percentage of first-ranked true motifs, our algorithm outperforms the Dmotif algorithm. One of the main aims of applying computational method in motif discovery is to narrow down the number of candidate motifs for further experimental validation. It is often desirable that an algorithm can report the true motifs at the highest rank. When such an algorithm is used to predict the candidate motifs, only very few motifs need to be tested. In addition, our algorithm also successfully found the binding site of the regulon *TBP*, which Dmotif algorithm failed to identify. The highest-ranked motif patten *TATAWAW* exactly matches the experimentally identified binding sites of *TBP*. In seven out of 22 cases, our algorithm assigns the experimentally determined binding sites a higher rank than the Dmotif algorithm, while in three out of 22 cases, Dmotif algorithm ranks them higher.

In two cases, both our proposed algorithm and Dmotif algorithm are unable to discover the annotated motif consensus. Table 2 lists the binding sites of regulon *UASPHR* for some promoter regions. Columns 1 and 3 are ORF ID, and columns 2 and 4 list the experimentally-mapped binding

sites in the corresponding upstream regions. The consensus pattern of the binding sites is determined to be CTTCTC according to SCPD. The sequence segments which have at most three mutations to the consensus pattern are underlined. As can be seen from the Table 2, the binding sites for *UASPHR* in different promoter regions clearly vary from each other, and only less than half of them have a pattern with less than four mutations from the consensus. The consensus pattern reported by our algorithm is RAT-GAAA. One possible explanation for the failure is that the large variations make the motif pattern too subtle to detect.

Table 2. Experimentally-mapped binding sites of UASPHR. The consensus of the binding sites is CTTCTC. The binding sites and consensus are both obtained from SCPD.

ORF ID	Binding site	ORF ID	Binding sites
YDL200C	GGAGGCCAGAAT	YDR217C	TGTT <u>ACTCCT</u> CG
YDR217C	TATA <u>CTTCTC</u> CG	YEL037C	GGTGGCGAAATT
YEL037C	TTT <u>CCTTCCT</u> CT	YER095W	CGTGGTGGGACC
YER095W	TTTT <u>TGGCA</u> CC	YER095W	CTTCTCTCTCT
YER142C	GGTGGCGATGAA	YER142C	TCTT <u>ATTCG</u> ACC
YER162C	CGTGGATGAAAC	YER162C	CGAGGCAGAATT
YGL058W	CAAGGAACAAAT	YIL066C	CTAGGTAGCAGA
YJL026W	CGAGGTCGCACA	YJR035W	AGTGAAGAAAA
YJR052W	GGAAGCAAAAAAT	YML032C	CGTGG <u>ATTCA</u> AC
YNL250W	CTCATCGCTTCT	YOR386W	CGAGGAAGCAGT
YOR386W	TTTT <u>CTTCCT</u> CG	YPL022W	GGAGGGAAGAAT
YPL153C	CGTGGGTAGAC	YBR114W	TGTT <u>ACTCCT</u> CG

5 Conclusions

Discovering motifs from biosequence data is a difficult task. In this paper, we have presented a new algorithm for finding motifs based on instance density. Experimental results show that our algorithm is capable of ranking the true motifs as top matches in 68% of the cases as compared with 45% of the cases with Dmotif algorithm. However, there are some cases that the algorithm failed to report true motifs as the top 10 motifs. The reason may lie in that there are considerable variations among binding sites presented in different promoter regions. This makes the distance between the motif and the set of sequences larger than the threshold. As a result, the true motif is not selected as candidate motif at all. This problem may be resolved by increasing the threshold value. However, this approach will also increase the number of candidate motifs selected for further tests. A direct consequence of this increment is a higher computational cost. On the other hand, because the candidate motifs are exhaustively enumerated, The complexity of this algorithm is $O(|A|^l)$ where $|A|$ is the size of the alphabet and l is the length of the motif. The PROJECTION algorithm by Buhler and Tompa[2] can be embedded in our algorithm to

reduce the number of candidate motifs being tested.

References

- [1] T. L. Bailey and C. Elkan. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning*, 21(1/2):51-80, 1995.
- [2] J. Buhler and M. Tompa. Finding motifs using random projections. *Proc. RECOMB*, 69-75, Montreal, Canada, 2001.
- [3] G. Z. Hertz and G. D. Stormo. Identifying DNA and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, 15:563-577, 1999.
- [4] U. Keich and P. Pevzner. Finding motifs in the twilight zone. *Proc. RECOMB*, 195-204, Washington, D.C., USA, 2002.
- [5] 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*, 262:208-214, 1993.
- [6] J. S. Liu, A. F. Neuwald, and C. E. Lawrence. Bayesian models for multiple local sequence alignment and Gibbs sampling strategies. *J. Am. Stat. Assoc.*, 90(432):1156-1170, 1995.
- [7] X. Liu, D. L. Brutlag and J. S. Liu. BIOPROSPER: Discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Proc. Pac. Symp. Biocomput.*, 127-38, 2001.
- [8] S. Sinha and M. Tompa. A statistical method for finding transcription factor binding sites. *Proc. ISMB*, 344-354, 2000.
- [9] S. Sinha. Discriminative motifs. *Proc. RECOMB*, 291-298, Washington, D.C., USA, 2002.
- [10] G. Thijs, K. Marchal, and Y. Moreau. A Gibbs Sampling method to detect over-represented motifs in the upstream regions of co-expressed genes. *Proc. RECOMB*, 305-312, Montreal, Canada, 2001.
- [11] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. Moor, P. Rouze and Y. Moreau. A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics*, 17(12):1113-1122, 2001.
- [12] J. Zhu and M. Q. Zhang. SCPD: a promoter database of the yeast *Saccharomyces cerevisiae*. *Bioinformatics*, 15(7/8):607-611, 1999.