REDUNDANCY ELIMINATION IN MOTIF DISCOVERY ALGORITHMS*

HENRY LEUNG AND FRINCIS CHIN

Department of Computer Science, The University of Hong Kong, Pokfulam, Hong Kong

Abstract: The problem of finding motifs in binding sites is very important to the understanding of gene regulatory networks. However, when predicting a set of motifs, existing algorithms suffer the problem of either predicting many redundant motifs (motifs with similar binding sites) or, at the other extreme, missing the hidden motif. In this paper, we formulate the Motif Redundancy Problem (MRP) to model this kind of problem and introduce an algorithm called RME (Redundancy Motif Elimination) for solving MRP. Experimental results on real biological data show that a standard EM-based motif discovery algorithm enhanced with RME has a better performance than the popular motif discovery algorithm MEME.

1 Introduction

A *gene* is a segment of DNA that is the blueprint for protein. In most cases, genes do not work alone; rather, they cooperate to produce different proteins for a particular function. In order to start the protein decoding process (*gene expression*), a molecule called *transcription factor* will bind to a short region (*binding site*) preceding the gene. One transcription factor can bind to the binding sites of several genes to cause these genes to co-express. These binding sites have similar patterns called *motifs*. Finding motifs and the binding sites from a set of DNA sequences, which represent the *promoter regions* of co-expressed genes, is a critical step for understanding the *gene regulatory network*.

In order to discover motifs, we must first have a model to represent the motif. There are two popular models: string representation [3,5-7,11,12,15,17,19,20,22-28] and matrix representation [1,2,8,9,13,14,16,18]. String representation is the most basic representation which uses a length-l string of symbols (or nucleotides) 'A', 'C', 'G' and 'T' to describe a motif. To improve the representation's descriptive power, wildcard symbols [5,22,26] can be introduced into the string to represent choices from a subset of symbols at a particular position (e.g. 'K' can denote 'G' or 'T'). Matrix representation further improves descriptive power. In the matrix model, motifs of length l are represented by *position weight matrices* (PWMs) or *position specific scoring matrices* (PSSMs) of size $4 \times l$ with the jth column of the matrix, which has four elements corresponding to the four nucleotides, effectively giving the occurrence probability of each of the four nucleotides at position j.

When discovering motif in matrix representation, researchers usually assume the motif matrix with the largest likelihood, calculated based on some probability model

* The research was supported in parts by the RGC grant HKU 7120/06E.

[1,13,14,16], is the hidden motif. However, motif discovery algorithms usually output a set of predicted motifs instead of a single motif with the highest likelihood because:

1. The input DNA sequences may contain binding sites of several transcription factors. Therefore, the algorithms should discover a motif for each of these transcription factors.
2. The motif matrix with the highest likelihood may not be a meaningful motif to the biologist and is considered only to be an over-represented pattern that occurs accidentally in the input sequences (noise), or alternatively, occurs in every part of the whole genome.
3. Biologists need to perform experiments to verify predicted motifs. Sometimes one experiment can be performed on several predicted motifs simultaneously in order to verify the correct one.

When motif discovery algorithms predict a set of motifs, they generally suffer the problem that some of the predicted motifs are very similar (*redundant motifs*) in the sense that they represent almost the same set of binding sites. For example, the motif matrices (predicted for AS-CT3) shown in Table 1, with M_1 being the hidden motif, are very similar:

Table 1. Example of similar motifs

Motif	Log Likelihood	PSSM	Binding sites pattern
M_1	-9833.12	A 0.21 0.49 0.19 0.14 0.12 0.17 C 0.51 0.27 0.13 0.20 0.19 0.19 G 0.16 0.07 0.23 0.48 0.47 0.19 T 0.11 0.17 0.44 0.16 0.22 0.45	CAGGTG [seq 1:355-360] CAGGTG [seq 2:336-341]
M_2	-9833.17	A 0.48 0.23 0.19 0.17 0.14 0.18 C 0.25 0.47 0.13 0.21 0.22 0.49 G 0.14 0.14 0.49 0.53 0.25 0.21 T 0.13 0.16 0.18 0.08 0.39 0.13	CAGGTG [seq 1:355-360] CAGGTG [seq 2:336-341] CAGGGG [seq 2:314-319]
M_3	-9832.36	A 0.65 0.00 0.00 0.06 0.00 0.01 C 0.03 0.00 0.13 0.00 0.01 0.02 G 0.27 0.89 0.79 0.00 0.82 0.21 T 0.04 0.10 0.08 0.94 0.17 0.76	AGGTGT [seq 1:356-361] AGGTGG [seq 2:337-341] AGGTGT [seq 2:31-36]

M_1 and M_2 have two binding sites in common. Although M_3 is different from M_1 and M_2, its first two binding sites overlap with the first two binding sites of M_1 and M_2 with a one base pair shift. Although these three motifs have high likelihoods, motif discovery algorithms should not output them all because they represent almost the same set of binding sites and would increase the size of output unnecessarily without meaningful benefit.

Some motif discovery algorithms [16] reduce the number of redundant motifs in the output by replacing a set of redundant motifs by a motif in that set. Other algorithms [14] (e.g. MEME) solve this problem by finding motifs one by one and by making, at each iteration, adjustments so as to reduce the probability that a binding site of an already-discovered motif is considered again. While this approach will help to reduce redundant motifs, accuracy may also suffer because the hidden motif might not be discovered if its binding sites happened to overlap with the binding sites of some previously discovered

motifs. This approach depends very much on the order of motifs being discovered. For example, consider the transcription factor "CF2-1" of fruit fly and the following discovered motifs arranged in the order of decreasing log-likelihood:

1	TTTTTTTT	log likelihood = -6399.45
2	GCGCCTGC	log likelihood = -6400.02
3	GCCCCCGC	log likelihood = -6403.37
	…	
19	GTTTTATT	log likelihood = -6409.44

The correct motif (ranked 19) will never be discovered in MEME if motif 1 is discovered first, nor, in those algorithms where redundant motifs are eliminated, because of its overlap with the first motif.

Moreover, the simple approach in which we allow for redundancy but limit the size of the output may also not work in situations such as that of the transcription factor "bcd" of fruit fly (see below) where the correct motif (ranked 281) is ranked far down the list.

1	CCCAACCC	log likelihood = -33744.8
2	CCAATCCC	log likelihood = -33753.4
3	CCCGATCC	log likelihood = -33755.8
	…	
281	TGGATTAG	log likelihood = -33870

In this paper, we introduce a novel way to select the "best" motifs among all possible motifs (redundant or otherwise) based on their likelihood of being the hidden motif and also their pair-wise redundancies. We first formulate the *Motif Redundancy Problem* (MRP) as the problem of picking a set of motifs for output such that accuracy will not be affected too much. However, since MRP is NP-hard, we cannot find the optimal solution of MRP in polynomial time unless P equals NP. Thus, we introduce a heuristic algorithm RME (Redundant Motif Elimination) to solve MRP. We show the usefulness of RME by comparing the performance of the popular software MEME against a simple EM algorithm enhanced by using RME to eliminate redundant motifs. We find that a simple EM algorithm can outperform MEME with the help of RME. Moreover, RME does successfully output the correct motif for the cases described above of "CF2-1" and "bcd".

This paper is organized as follows. In Section 2, we briefly describe how to calculate the likelihood of a matrix being the hidden motif. In Section 3, we introduce MRP. The heuristic algorithm RME for solving MRP is described in Section 4. Experimental results on real biological data comparing are given in Section 5, followed by concluding remarks in Section 6.

2 Maximizing Likelihood

Existing algorithms using PSSM matrix representation discover the motif matrix with the maximum likelihood of being the hidden motif based on the finite mixture model [1]. Conceptually, they break up the input sequences into length-l (overlapping) substrings. For example, a length-n sequence can be broken up into $w = n - l + 1$ length-l substrings. Let $X = (X_1, X_2, \ldots, X_w)$ be all length-l substrings in the input where each substring can occur more than once in X if the same pattern appears in more than one position. The finite mixture model assumes that each substring in X belongs to either a background (non-motif) substring or an instance of the hidden motif. The prior probability that a substring belongs to the background substrings (generated according to the background probability) is λ_b and the prior probability that a substring belongs to binding sites (generated according to the hidden matrix) is $1 - \lambda_b$.

Let $Z = (Z_1, Z_2, \ldots, Z_w)$ be the missing data that determines whether X_i is generated according to the background probability B or the hidden matrix M.

$$Z_i = \begin{cases} 1 & X_i \text{ is generated according to } B \\ 0 & X_i \text{ is generated according to } M \end{cases}$$

The conditional probability that a substring X_i is generated according to the background probability $B = (b(\mathrm{A}), b(\mathrm{C}), b(\mathrm{G}), b(\mathrm{T}))$ is

$$p(X_i \mid Z_i = 1, B) = \prod_{j=1}^{l} b(X_i[j])$$

where $X_i[j]$ is the jth nucleotide in X_i.

The conditional probability that a substring X_i is generated according to a $4 \times l$ hidden matrix M is

$$p(X_i \mid Z_i = 0, M) = \prod_{j=1}^{l} M(X_i[j], j)$$

where $M(\alpha, j)$ is the probability that nucleotide α occurs at the jth position of an instance of M. Note that $\sum_\alpha M(\alpha, j) = 1$.

By the assumption of independent substrings X, the joint conditional density of X and Z being generated according to the finite mixture model with parameters B, M and λ_b is

$$\begin{aligned} & p(X, Z \mid B, M, \lambda_b) \\ = & \prod_{i=1}^{w} \left(p(X_i, Z_i \mid B, M, \lambda_b) \right) \\ = & \prod_{i=1}^{w} \left(p(X_i \mid Z_i, B, M, \lambda_b)\, p(Z_i \mid B, M, \lambda_b) \right) \\ = & \prod_{i=1}^{w} \left(\left[\lambda_b \prod_{j=1}^{l} b(X_i[j]) \right]^{Z_i} \left[(1-\lambda_b) \prod_{j=1}^{l} M(X_i[j], j) \right]^{1-Z_i} \right) \end{aligned}$$

The likelihood of a particular B, M, λ_b being the hidden parameters of the finite mixture model given the joint distribution of the substring X and the missing data Z is defined as

$$L(B,M,\lambda_b \mid X,Z) = p(X,Z \mid B,M,\lambda_b) \tag{1}$$

So, the log of the likelihood, or *log likelihood*, is therefore

$$\begin{aligned} &\log L(B,M,\lambda_b \mid X,Z) \\ &= \sum_{i=1}^{w} \left\{ Z_i \left[\log(\lambda_b) + \sum_{j=1}^{l} \log\big(b(X_i[j])\big) \right] + (1-Z_i) \left[\log(1-\lambda_b) + \sum_{j=1}^{l} \log\big(M(X_i[j],j)\big) \right] \right\} \end{aligned} \tag{2}$$

The goal of existing algorithms using matrix representation is to discover the B, M, λ_b with the maximum likelihood (or log likelihood).

3 The Motif Redundancy Problem

In practice, biologists want to get a set of predicted motif matrices with high likelihood instead of a single matrix with the highest likelihood (as discussed in the introduction). However, existing motif discovery algorithms either allow the output of many redundant motifs (i.e. motifs representing almost the same set of binding sites), which makes the output size unnecessarily large, or try to eliminate redundant motifs and in the process risk also eliminating the hidden motif. We introduce the *Motif Redundancy Problem* (MRP) to reduce the size of the output with the least reduction in accuracy.

3.1 The Motif Redundancy Problem

Assume the positions of the planted binding sites of the hidden matrix M^* are known. The accuracy of a predicted motif with matrix M is measured by its *score* $s(M,M^*)$ expressed as follows [3]:

$$s(M,M^*) = \frac{|\text{sites of } M \cap \text{sites of } M^*|}{|\text{sites of } M \cup \text{sites of } M^*|} \tag{3}$$

We say a planted binding site at position $[x, x + l - 1]$ is *correctly predicted* if that planted binding site overlaps with at least one predicted site $[y, y + l - 1]$, i.e. $[x, x + l - 1] \cap [y, y + l - 1]$ is non-empty. The score $s(M,M^*)$ is in the range of [0,1]. When all the planted binding sites are correctly predicted without any mis-prediction, score = 1. When no planted binding site is predicted correctly, score = 0. Since biologists will select the best motif as the hidden motif, the *accuracy of a motif discovery algorithm* can be measured by the maximum score obtained from its set of predicted motifs, i.e. $\max\{s(M,M^*) \mid M \in S\}$ where S is the set of motif matrices predicted by the algorithm. Note that accuracy increases with the size of output of any motif discovery algorithm.

3.2 Motif Redundancy Problem

Given a set S of motif matrices M_i (where i = 1, …, q) and their binding sites and corresponding likelihood $L_i = L(B, M_i, \lambda_b \mid X, Z)$. Assuming the hidden motif is one of the q matrices in S, we can estimate the probability $P(M_j \mid X)$ that M_j is the hidden matrix of the data set X by its likelihood $L_i = L(B, M_i, \lambda_b \mid X, Z)$.

Since $L_i = L(B, M_i, \lambda_b \mid X, Z) = p(X, Z \mid B, M_i, \lambda_b) \propto P(M_i \mid X)$, we have

$$P(M_j \mid X) = \frac{L_j}{\sum_{k=1}^{q} L_k} \tag{4}$$

As the score of M_i is $s(M_i, M_j)$ if M_j is the hidden matrix, the *expected score* of M_i is

$$E(M_i) = \sum_{j=1}^{q} s(M_i, M_j) P(M_j \mid X)$$

Given a set S of motif matrices, their binding sites and likelihoods, $E(S)$ can be calculated with Eq (1), (3) and (4). That is,

$$E(S) = \sum_{j=1}^{q} \max_{M \in S} \{s(M, M_j)\} P(M_j \mid X) \tag{5}$$

The *Motif Redundancy Problem* (MRP) is defined as follows:

Given a set of input sequences, a set of motif matrices M_i (where i = 1, …, q), their binding sites and corresponding likelihood $L_i = L(B, M_i, \lambda_b \mid X, Z)$, find a subset $S \subseteq \{M_i, i = 1, ..., q\}, \mid S \mid = m$ such that $E(S)$ is maximized.

Note that a set of redundant motifs usually has a lower score on average, because $s(M, M_j)$ of each redundant motif M is almost the same and there will be many motifs M_j not in S with a low $\max\{s(M, M_j) \mid M \in S\}$ value. This means there is a higher chance of missing the hidden motif. Therefore, a set S with the largest expected score $E(S)$ tends to contain non-redundant motifs.

4 Algorithm

We can show that MRP is a NP-hard problem by transforming the Set Covering Problem to MRP (shown in the Appendix). Therefore, it is not possible to find a polynomial time algorithm to solve MRP unless P equals NP. We apply a heuristic algorithm RME (which stands for *Redundancy Motif Elimination*) to solve MRP. RME finds the subset S of motifs with large expected score $E(S)$ by selecting motifs that give the largest increase in $E(S)$ one by one until the size of S is m. This is essentially a greedy approach. Although RME is simple, experimental results (see Section 5) on real biological data show that it works well in practice. Algorithm RME is shown in below.

Algorithm RME

1. $S = \{\arg\max_{M_i} E(M_i)\}, i = 1,..., q$
2. For i = 2 to m
3. $S = \{\arg\max_{M_j} E(S \cup M_j)\}, j = 1,..., q$
4. output S

First we begin with S containing the motif matrix with the highest expected score $E(M_i)$. At each step, we add a new motif matrix to S such that the new expected score $E(S)$ has the largest increase in value.

In order to illustrate how RME works, let us consider the following example.

Example. Suppose $\{M_1, M_2, M_3\}$ is a set of three predicted matrices with 0.45, 0.35 and 0.2 as their corresponding $P(M_j \mid X)$ where M_1 and M_2 are two redundant motifs. Thus, $s(M_1, M_2) = s(M_2, M_1) = 0.8$ and $s(M_3, M_1) = s(M_3, M_2) = s(M_1, M_3) = s(M_2, M_3) = 0.2$. RME would first choose M_1 since $E(M_1) = 1 * 0.45 + 0.8 * 0.35 + 0.2 * 0.2 = 0.77$ (by Eq (5)) has the highest expected score. If $m = 2$, the second motif matrix will be M_3 instead of M_2 even though M_2 has a higher likelihood since $E(\{M_1, M_2\}) = 1 * 0.45 + 1 * 0.35 + 0.2 * 0.2 = 0.84$ and $E(\{M_1, M_3\}) = 1 * 0.45 + 0.8 * 0.35 + 1 * 0.2 = 0.93$.

5 Experimental Results

We have implemented the standard EM algorithm [14] for discovering motifs and RME on C++ and have performed experiments on real biological data for fruit fly (Drosophila) and yeast from the database TRANSFAC (http://www.gene-regulation.com) and SCPD (http://rulai.cshl.edu/SCPD/). For each transcription factor, we searched for all genes regulated by that transcription factor and used the 450 base pairs (bp) upstream and 50 bp downstream from the transcriptional start site of these genes as the input sequences.

MEME [1], a popular motif discovery program (which is based on a more complicated EM-algorithm), was compared to the performance of a standard EM algorithm enhanced with RME and the standard EM algorithm without RME. Each of the three algorithms predicted 30 motifs with length equal to the published motif of the corresponding transcription factor. The standard EM algorithm first generated 300 predicted motifs and, with and without RME, 30 motifs were picked as output. The performance of these algorithms on each transcription factor was measured by the following formula for accuracy and is shown in Tables 2 (fruit fly) and Table 3 (yeast) below.

$$\text{accuracy} = \frac{|\text{predicted sites} \cap \text{published sites}|}{|\text{predicted sites} \cup \text{published sites}|}$$

In the 47 experiments of the fruit fly, all three algorithms failed to predict any published binding site correctly in 4 data sets (which are not shown in Table 2). For the remaining 43 data sets, the standard EM algorithm with RME has better performance (i.e. higher score) in 28 data sets and equal performance (i.e. the same score) in 4 data sets,

while MEME had a better performance in 10 data sets and the standard EM algorithm without RME has better performance in 1 data set only. Moreover, the average score of the standard EM algorithm with RME is 0.2787, which is higher than the average score of MEME of 0.1934 and the average score of the standard EM algorithm without RME of 0.0873.

Table 2. Experimental results on real biological data for transcription factors of fruit fly (Drosophila) in TRANSFAC

Factor Name	*l*	MEME	EM	RME	Factor Name	*l*	MEME	EM	RME
Ac	8	**1**	0.0053	0.4	Exd	20	0	0	**0.3333**
adf-1	11	**0.1667**	0.0769	0.15	Ftz	12	0.1724	0.1351	**0.2424**
antp	7	0	0.0476	**0.2**	FTZ-F1	7	0.3333	0	**0.6667**
AP-1	9	0	0	**0.1429**	GAGA	11	0.0870	0.1	**0.2222**
AS-CT3	6	**1**	0.0435	**1**	GCM	13	0.25	0.4286	**0.5**
bcd	8	0.2941	0.0130	**0.4375**	H	10	0	0.1667	**0.25**
BEAF-32B	5	0	**0.7143**	0.25	Hb	10	0.1429	0.1875	**0.2941**
Bfactor	4	**0.1**	0	0	HSTF	15	**0.1667**	0	**0.1667**
CF1	9	0	0	**0.2**	Kr	10	**0.3077**	0.0526	0.2778
CF2-I	8	0	0	**0.1667**	Sc	8	**0.5**	0.0526	0.4
Ci	9	0.2857	0.0909	**0.3636**	Sn	13	**0.2857**	0	0.2105
Cut	7	0	0	**0.1667**	Su_Hw	12	0	0.1	**0.1**
D_MEF2	10	0	0	**0.1667**	TAB	15	**0.3333**	0.1111	0.1333
D1	11	0.0870	0.0909	**0.1379**	TBP	7	0.25	0	**0.4**
Da	6	0	0	**0.3333**	TII	8	0.0769	0.0938	**0.1304**
DREF	14	**0.1429**	0	0	Ttk69k	8	**0.3333**	0.2222	**0.3333**
dri	10	0	0	**0.1429**	Ubx_a	19	0	0.0833	**0.25**
DTF-1	6	0.25	0.1111	**0.75**	Zen-1	8	0.1	0.1538	**0.1579**
E74A	17	**0.4**	0.1143	0.2941	Zen-2	8	0.1111	0.1111	**0.25**
EcR	7	**0.3333**	0	**0.3333**	Zeste	11	0.1026	0.1290	**0.3191**
Elf-1	8	**0.5**	0.1	0.1429	Zeste_b	11	0.1026	0.1290	**0.3191**
En	7	0.1	0.0909	**0.25**					
Average score							0.1934	0.0873	**0.2787**
Number of times getting the best performance							14	1	**32**

In the experiments, the performance of the standard EM algorithm was much improved with the application of RME (except BEAF-32B). Instead of selecting 30 redundant motifs with the highest score generated by the EM algorithm, RME selected 30 non-redundant motifs, which increased the probability of predicting the hidden motifs. Although the standard EM algorithm has a worse performance than the sophisticated MEME in most cases, it outperformed MEME after applying RME.

In the 32 experiments of yeast, all three algorithms failed to predict any published binding site correctly in 1 data set (which is not shown in Table 3). For the remaining 31 data sets, the standard EM algorithm with RME had better performance (higher score) in 15 data sets and the same score in 6 data sets, while MEME had better performance in 10 data sets. The average score of the standard EM algorithm with RME was 0.4845, which was slightly lower than the average score of MEME of 0.4995 but much higher than the average score of the standard EM algorithm of 0.2030.

Table 3. Experimental results on real biological data for transcription factors of yeast in SCPD

Factor Name	*l*	MEME	EM	RME	Factor Name	*l*	MEME	EM	RME
13nt	13	**0.75**	0.375	**0.75**	GFI	13	0.2	0	**0.2222**
ABF1	13	0.0870	0.0082	**0.125**	HAP1	12	**0.5833**	0	0.5625
ACE2	6	0.3333	0.3	**0.6**	HAP2	7	**0.3333**	0.0833	**0.3333**
ADR1	5	**0.6667**	0.3333	**0.6667**	HSE_2	8	**0.7**	0	0.6
AP1	7	**1**	0	0.25	IRE	32	**1**	0.1053	0.5
BAS1	7	0.3889	0	**0.4444**	LEU	10	0.6667	0.3333	**1**
BAS2	6	0.25	0	**0.3333**	MAT2	9	0.3333	0.0690	**0.4375**
CCBF	7	**0.9375**	**0.9375**	**0.9375**	MCM1	5	**0.4590**	0	0
CPF1	7	**0.6667**	0.1818	0.6	MIG1	12	**0.3333**	0.0164	0.2174
CSRE	12	0.3333	0.0731	**0.3636**	NBF	9	0.5	0.375	**0.5714**
CURE	7	**0.6667**	0.5714	**0.6667**	SFF	10	**0.2**	0	0.1875
GAL4	17	**0.8666**	0.8235	0.8235	SWI5	6	0.4444	0	**0.5714**
GATA	6	0.3913	**0.45**	**0.45**	UASCAR	11	0.25	0	**0.5**
GCFAR	6	0.5714	0.5714	**0.8**	UASGABA	19	0.4	**0.6667**	**0.6667**
GCN4	6	0.3333	0.0222	**0.3571**	UESPHR	9	**0.6667**	0	0.4285
GCR1	5	**0.1818**	0	0.0526					
Average score							**0.4998**	0.2030	0.4845
Number of times getting the best performance							16	3	**21**

As shown by the results given in Tables 2 and 3, the standard EM algorithm without RME had a worse performance than MEME in all cases (except GATA and UASGABA). However, the standard EM algorithm had a similar performance to MEME after applying RME. This indicates that algorithm RME can indeed improve the performance of motif discovery algorithms.

6 Concluding Remarks

Many motif discovery algorithms use heuristics to search for motif matrices according to some performance criteria. In most cases, many redundant (similar) motif matrices will be found based on local optimal values. We introduce the motif redundancy problem (MRP) so as to find the best fixed-size output with highest accuracy. Even though MRP is NP-complete, we have demonstrated in this paper that a simple greedy approach (RME) can already bring an improvement in the accuracy of the output. However, this is only a preliminary result, and there are at least two other approaches one could take to make further improvements: (1) improving the heuristic used in solving MRP; and (2) the elimination of the assumption that the hidden motif is among one of the predicted motif. We would include these improvements in our future paper.

Appendix

In this section, we will show how to reduce the Set Covering Problem (SC) to the Motif Redundancy Problem (MRP). The Set Covering Problem is defined as follows:

Given a finite set $X = \{x_1, \ldots, x_n\}$ and a family F of subsets $X_1, \ldots, X_p$ such that every elements x_i in X belongs to at least one subsets in the family. We want to determine whether there is a subset $C \subseteq F, |C| = k$ such that $X = \cup_{X_i \in C} X_i$.

We reduce SC to MRP in the following manner. Let $W = p + n$. We construct $W + pW$ motifs. The first p motifs (set P) represent the p subsets $X_1, \ldots, X_p$, the next n motifs (set N) represent the n elements $x_1, \ldots, x_n$ in set X and the last pW are dummy motifs (set D).

Each motif has equal likelihood $1/(W + pW)$ and has exactly $W + pW$ binding sites. There are two ways in which a motif can have exactly one binding sites overlapped with another motif:

1. The motif representing X_i has one binding site overlapping with the motif representing x_j if and only if $x_j \in X_i$.
2. The motif representing X_i has one binding site overlapping with W distinct dummy motifs.

Except as a result of applying the above two rules, no other binding-site overlaps are allowed.

In particular, the $W + pW$ binding sites for each motif are constructed in the following manner. Each binding site is of length-(W+pW) DNA sequence with nucleotide 'T' at every position except the yth position in the case of yth motif and the zth position if the binding site of this motif is meant to overlap with that of the yth motif (according to one of the above two rules). Nucleotide 'A' appears at such (yth and zth) positions instead.

SC can be reduced to MRP by finding a subset S of motifs, $|S| = k$ such that $E(S)$ is maximized. There is a solution for SC if and only if

$$E(S) = \frac{1}{W + pW}\left(k + (n + kW)\frac{1}{2(W + pW) - 1}\right)$$

References

1. T. Bailey and C. Elkan. Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Machine Learning*, 21:51—80, 1995.
2. Y. Barash, G. Bejerano and N. Friedman. A simple hyper-geometric approach for discovering putative transcription factor binding sites. *WABI*, 278—293, 2001.
3. J. Buhler and M. Tompa. Finding motifs using random projections. *RECOMB*, 69—76, 2001.
4. M.L. Bulyk, P.L.F. Johnson and G.M. Church. Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nuc. Acids Res.*, 30:1255—1261, 2002.
5. F. Chin and H. Leung. An Efficient Algorithm for String Motif Discovery. *APBC*, 79—88, 2006.
6. F. Chin and H. Leung. An Efficient Algorithm for the Extended (l,d)-Motif Problem With Unknown Number of Binding Sites. *BIBE*, 11—18, 2005.

7. F. Chin and H. Leung. Voting Algorithms for Discovering Long Motifs. *APBC*, 261—271, 2005.
8. F. Chin, H. Leung, S.M. Yiu, T.W. Lam, R. Rosenfeld, W.W. Tsang, D. Smith and Y. Jiang. Finding Motifs for Insufficient Number of Sequences with Strong Binding to Transcription Factor. *RECOMB*, 125—132, 2004.
9. G.Z. Hertz and G.D. Stormo. Identification of consensus patterns in unaligned dna and protein sequences: a large-deviation statistical basis for penalizing gaps. *The Third International Conference on Bioinformatics and Genome Research*, 201—216, 1995
10. J.D. Hughes, P.W. Estep, S. Tavazoie and G.M. Church. Computational identification of cis-regulatory elements associated with groups of functionally related genes in Saccharomyces cerevisiae. *Journal of Molecular Biology*, 296(5):1205—14, 2000.
11. U. Keich and P. Pevzner. Finding motifs in the twilight zone. *RECOMB*, 195—204, 2002.
12. S. Kielbasa, J. Korbel, D. Beule, J. Schuchhardt and H. Herzel. Combining frequency and positional information to predict transcription factor binding sites. *Bioinformatics*, 17:1019—1026, 2001.
13. 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.
14. C. Lawrence and 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.
15. H. Leung and F. Chin. Algorithms for Challenging motif problems. *JBCB*, 43—58, 2005.
16. H. Leung and F. Chin. Finding Exact Optimal Motif in Matrix Representation by Partitioning. *Bioinformatics*, 22:ii86—ii92, 2005.
17. H. Leung and F. Chin. Generalized Planted (*l*,*d*)-Motif Problem with Negative Set**.** *WABI*, 264—275, 2005.
18. H. Leung, F. Chin, S.M. Yiu, R. Rosenfeld and W.W. Tsang. Finding Motifs with Insufficient Number of Strong Binding Sites. *Jour. Comp. Biol.*, 12(6):686—701, 2005.
19. M. Li, B. Ma and L. Wang. Finding similar regions in many strings. *Journal of Computer and System Sciences*, 65:73—96, 2002.
20. S. Liang. cWINNOWER Algorithm for Finding Fuzzy DNA Motifs. *Computer Society Bioinformatics Conference*, 260—265, 2003.
21. T.K. Man and G.D. Stormo. Non-independence of Mnt repressor-operator interaction determined by a new quantitative multiple fluorescence relative affinity (QuMFRA) assay. *Nuc. Acids Res.*, 29:2471—2478, 2001.
22. G. Pesole, N. Prunella, S. Liuni, M. Attimonelli and C. Saccone. Wordup: an efficient algorithm for discovering statistically significant patterns in dna sequences. *Nucl. Acids Res.*, 20(11):2871—2875, 1992.
23. 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*, 269—278, 2000.

24. S. Rajasekaran, S. Balla and C.H. Huang. Exact algorithms for planted motif challenge problem. *APBC*, 249—259, 2005.
25. S. Sinha. Discriminative motifs. *The Sixth Annual International Conference on Computational Biology*, 291—298, 2002.
26. S. Sinha and M. Tompa. A statistical method for finding transcription factor binding sites. *The Eighth International Conference on Intelligent Systems for Molecular Biology*, 344—354, 2000.
27. K.T. Takusagawa and D.K. Gifford. Negative information for motif discovery. *PSB*, 360—371, 2004.
28. M. Tompa. An exact method for finding short motifs in sequences with application to the ribosome binding site problem. *The Seventh International Conference on Intelligent Systems for Molecular Biology*, 262—271, 1999.
29. Y. Xing, J.D. Fikes and L. Guarente. Mutations in yeast HAP2 HAP3 define a hybrid CCAAT box binding domain. *EMBO Journal*, 12:4647—4655, 1993.
30. X. Zhao, H. Huang and T.P. Speed. Finding Short DNA Motifs Using Permuted Markov Models. *RECOMB*, 68—75, 2004.