DNA Motif Representation with Nucleotide Dependency

Francis Chin and Henry C.M. Leung

Abstract—The problem of discovering novel motifs of binding sites is important to the understanding of gene regulatory networks. Motifs are generally represented by matrices (position weight matrix (PWM) or position specific scoring matrix (PSSM)) or strings. However, these representations cannot model biological binding sites well because they fail to capture nucleotide interdependence. It has been pointed out by many researchers that the nucleotides of the DNA binding site cannot be treated independently, for example, the binding sites of zinc finger in proteins. In this paper, a new representation called Scored Position Specific Pattern (SPSP), which is a generalization of the matrix and string representations, is introduced, which takes into consideration the dependent occurrences of neighboring nucleotides. Even though the problem of discovering the optimal motif in SPSP representation is proved to be NP-hard, we introduce a heuristic algorithm called SPSP Finder, which can effectively find optimal motifs in most simulated cases and some real cases for which existing popular motif-finding software, such as Weeder, MEME, and AlignACE, fail.

Index Terms—Computing methodologies, pattern recognition, design methodology, pattern analysis.

1 Introduction

A *gene* is a segment of the DNA that is the blueprint for protein. In most cases, genes seldom 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 kind of transcription factor can bind to the binding sites of several genes to cause these genes to coexpress. These binding sites have similar patterns called *motifs*. Discovering novel motifs of unknown transcription factors and the binding sites from a set of DNA sequences is a critical step for understanding the *gene regulatory network*.

In order to discover motifs of unknown transcription factors, we must first have a model to represent motifs. There are two popular models: string representation [4], [6], [7], [8], [13], [14], [17], [20], [22], [23], [25], [26], [27], [28], [29], [30], [31], [32], [33] and matrix representation [1], [2], [9], [11], [15], [16], [18], [19], [21]. 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 [6], [26], [31] can be introduced into the string to represent choice from a subset of symbols at a particular position (for example, "K" can denote "G" or "T"). Matrix representation further improves descriptive power. In the matrix representation, motifs of length l are represented by *position weight matrices* (PWMs) or *position specific scoring matrices* (PSSMs) of size $4 \times l$ with the four entries in the jth column of the matrix, effectively giving the occurrence probabilities of the four nucleotides at position j. Although matrix representation appears superior, the solution space for PWMs and PSSMs, which consists of $4l$ real numbers is infinite in size, and there are many local optimal matrices, thus, algorithms generally either produce a suboptimal motif matrix [1], [2], [9], [15], [16], [21] or take too long to run when the motif is longer than 10 bp [19].

As it turns out, the string and the matrix representations share an important common weakness: They assume the occurrence of each nucleotide at a particular position of a binding site is independent of the occurrence of nucleotides at other positions. This assumption does not represent the true picture. According to Bulyk et al. [5], analysis of wild-type and mutant Zif268 (Egr1) zinc fingers gives compelling evidence that nucleotides of transcription factor binding sites should not be treated independently, and a more realistic motif representation should be able to describe nucleotide interdependence. Man and Stormo [24] have arrived at a similar conclusion in their analysis of *Salmonella* bacteriophage repressor Mnt: They found that interactions of Mnt with nucleotides at positions 16 and 17 of the 21 bp binding site are in fact not independent.

When the positions of binding sites are known, we may represent the motif by Hidden Markov Model (HMM) [36], Bayesian network [3], or enhanced PWM [10], which can overcome the above weakness. However, these models cannot be easily extended to discover novel motifs especially when the number of coexpressed genes is small (say, less than 10). It is because the input data does not contain enough information for deriving the hidden motif and the above models usually overfit the input data. Hence, they are far less popular representations.

In this paper, we introduce a new motif representation called *Scored Position Specific Pattern* (SPSP), which has the following advantages:

• *The authors are with the Department of Computer Science, The University of Hong Kong, Pok Fu Lam Road, Hong Kong 852. E-mail: {chin, cmleung2}@cs.hku.hk.*

Manuscript received 2 May 2006; revised 5 Oct. 2006; accepted 13 May 2007; published online 7 June 2007.
For information on obtaining reprints of this article, please send e-mail to: tcbb@computer.org, and reference IEEECS Log Number TCBB-0103-0506.
Digital Object Identifier no. 10.1109/TCBB.2007.70220.

1. *Better representation.* SPSP can describe the interdependence between neighboring nucleotides with similar number of parameters as string and matrix representations.
2. *Generalization of string and matrix representations.* These two commonly used representations are special cases of the SPSP representation. Thus, SPSP representation can model more motifs than these two representations.
3. *Computationally feasibility.* Finding the optimal motif in SPSP representation, for some restricted cases, is more feasible than finding the optimal PWM or PSSM.

This paper tackles a "restricted" motif discovering (RMD) problem based on the SPSP representation. Although this is a restricted problem, it can model all motifs in string representation and most motifs in matrix representation. Because this restricted problem is NP-complete (proof shown in the Appendix), we introduce a heuristic algorithm called *SPSP Finder*, which can find the optimal SPSP motifs in most simulated cases and some real cases, for which Weeder [25], MEME [16], and AlignACE [12] fail.

This paper is organized as follows: In Section 2, we describe the SPSP representation, the corresponding motif problem and its restricted version in detail. In Section 3, we introduce the heuristic algorithm SPSP Finder. Experimental results on simulated data and real biological data comparing SPSP Finder with some popular software are given in Section 4, followed by concluding remarks in Section 5.

2 Scored Position Specific Pattern (SPSP)

Consider the wildcard-augmented string representation with 15 symbols representing all combinations of the four nucleotides "A," "C," "G," and "T." For example, the wildcard symbol "Y" represents "C" or "T" and wildcard symbol "S" represents "C" or "G." Consider the motif for the transcription factor HAP2 [34], which exists as a heterotrimeric complex with the HAP3 and HAP4 proteins. The HAP2/3/4 complex binds to the patterns "CCAATCA," "CCAATGA," or "CCAACCA." We can represent the motif by "CCAAYSA" with two wildcard symbols. In fact, we may also represent "CCAAYSA" as follows:

$$(\mathrm{C})(\mathrm{C})(\mathrm{A})(\mathrm{A})\begin{pmatrix}\mathrm{C}\\ \mathrm{T}\end{pmatrix}\begin{pmatrix}\mathrm{C}\\ \mathrm{G}\end{pmatrix}(\mathrm{A}).$$

However, this representation has the problem that the pattern "CCAACGA" is also considered as a binding site (false positive). In order to prevent the inclusion of false positive patterns, we replace the substring "YS" by a set of length-2 patterns, i.e.,

$$(\mathrm{C})(\mathrm{C})(\mathrm{A})(\mathrm{A})\begin{pmatrix}\mathrm{TC}\\ \mathrm{TG}\\ \mathrm{CC}\end{pmatrix}(\mathrm{A}) \text{ or } (\mathrm{CCAA})\begin{pmatrix}\mathrm{TC}\\ \mathrm{TG}\\ \mathrm{CC}\end{pmatrix}(\mathrm{A}).$$

The SPSP representation uses such an idea to represent motifs. Based on this SPSP representation, our algorithm can find the motif and binding sites of HAP2, whereas the other software fails to do so. The formal definition of SPSP is described in Section 2.1.

2.1 Formal Definition of Pattern Sets Representation

A set of length-l binding site patterns can be described by a *SPSP* representation P, which contains $c\,(c \le l)$ sets of patterns P_i, $1 \le i \le c$, where each set of patterns P_i contains length-l_i patterns $P_{i,j}$ of symbols "A," "C," "G," and "T," and $\sum_i l_i = l$. Each length-l_i pattern $P_{i,j}$ is associated with a score $s_{i,j}$ that represents the "closeness" of a pattern to be a binding site, that is, the lower the score, the more likely that the pattern is a binding site. The score of a length-l string $\sigma = \sigma_1\sigma_2\ldots\sigma_c$, where $|\sigma_i| = l_i$, $1 \le i \le c$ with respect to P can be defined as follows:

$$\mathrm{score}(\sigma, P) = \sum_{i=0}^{c}\begin{cases} s_{i,j} & \exists j, P_{i,j} = \sigma_i \\ \infty & \text{otherwise.}\end{cases}$$

A string σ is a *binding site* with respect to an SPSP motif P if and only if $\mathrm{score}(\sigma, P)$ is no more than some predefined threshold α.

For example, consider the following SPSP representation for the length-11 binding sites of the transcription factor CSRE [37], which activates the gluconeogenic structural genes:

$$P = (\mathrm{CGGA})\begin{pmatrix}\mathrm{TGA}\\ \mathrm{TAA}\\ \mathrm{CGG}\end{pmatrix}(\mathrm{A})\begin{pmatrix}\mathrm{A}\\ \mathrm{T}\end{pmatrix}(\mathrm{GG}) \text{ and}$$

$$\{s_{i,j}\} = (-\log(1))\begin{pmatrix}-\log(0.5)\\ -\log(0.3)\\ -\log(0.2)\end{pmatrix}(-\log(1))\begin{pmatrix}-\log(0.7)\\ -\log(0.3)\end{pmatrix}(-\log(1)).$$

Note that the score $s_{i,j}$ is the negative of the logarithm of the occurrence probability of the corresponding pattern $P_{i,j}$. The score of the length-11 string $\sigma = ''\mathrm{CGGATAAAAGG}''$ with $\sigma_1 = ''\mathrm{CGGA}''$, $\sigma_2 = ''\mathrm{TAA}''$, $\sigma_3 = ''\mathrm{A}''$, $\sigma_4 = ''\mathrm{A}''$, and $\sigma_5 = ''\mathrm{GG}''$ can be calculated as

$$-\log(1) - \log(0.3) - \log(1) - \log(0.7) - \log(1) = -\log(0.21).$$

On the other hand, the score of $\sigma = ''\mathrm{CTGATAAAAGG}''$ is ∞ as $\sigma_1 = ''\mathrm{CTGA}'' \notin P_1$. The scores of these strings represent the negative log likelihood of these strings being binding sites of P. A string with smaller score is more likely to be a binding site of P.

Based on the SPSP representation, we can define the Motif Discovering (MD) Problem as follows:

MD Problem. Given t length-n DNA sequences T, we want to find a motif M in SPSP representation (P and score $\{s_{i,j}\}$ satisfying certain properties) to maximize/minimize some target function calculated based on the scores of the binding sites of M in T.

The following will show that SPSP representation is a generalization of the string and matrix representations. By applying different target functions, we can discover motifs with different properties under a certain score scheme $\{s_{i,j}\}$.

1. Restricting $c = l$ (that means $l_i = 1, 1 \leq i \leq c = l$), the SPSP representation P is equivalent to a PWM or PSSM [1], [15], [16], [19]. Using the following probability matrix for transcription factor CSRE with threshold 0.04 as an example.

$$\begin{matrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{matrix} \begin{pmatrix} 0 & 0 & 0 & 1.0 & 0 & 0.3 & 0.7 & 1.0 & 0.7 & 0 & 0 \\ 1.0 & 0 & 0 & 0 & 0.3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1.0 & 1.0 & 0 & 0 & 0.7 & 0.3 & 0 & 0 & 1.0 & 1.0 \\ 0 & 0 & 0 & 0 & 0.7 & 0 & 0 & 0 & 0.3 & 0 & 0 \end{pmatrix}.$$

It is equivalent to the following SPSP representation:

$$P = (\mathrm{C})(\mathrm{G})(\mathrm{G})(\mathrm{A})\begin{pmatrix} \mathrm{C} \\ \mathrm{T} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{G} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{G} \end{pmatrix}(\mathrm{A})\begin{pmatrix} \mathrm{A} \\ \mathrm{T} \end{pmatrix}(\mathrm{G})(\mathrm{G}) \text{ and}$$

$$\{s_{i,j}\} = (0)(0)(0)(0)\begin{pmatrix} -\log(0.3) \\ -\log(0.7) \end{pmatrix}\begin{pmatrix} -\log(0.3) \\ -\log(0.7) \end{pmatrix}\begin{pmatrix} -\log(0.7) \\ -\log(0.3) \end{pmatrix}(0)\begin{pmatrix} -\log(0.7) \\ -\log(0.3) \end{pmatrix}(0)(0),$$

with threshold $\alpha = -\log(0.04)$. Note that

$$-\log(1.0) = 0.$$

In order to find a set of binding sites with the minimum negative log likelihood, the MD problem is to find P and $\{s_{i,j}\}$ such that for $1 \leq i \leq c = l$, $s_{i,j} = -\log(p_{i,j})$ with $\sum_j p_{i,j} = 1$ so as to minimize the target function $\sum_\sigma [\mathrm{score}(\sigma, P) + l \log(0.25)]$ for all binding site σ (i.e., with $\mathrm{score}(\sigma, P) \leq \alpha(\mathrm{threshold})$).
2. Restricting $c = l$, $s_{i,j} = 0$ or 1, $\sum_j s_{i,j} = 3$, and $\alpha = d$, the SPSP representation P is equivalent to a string representation [4], [8], [22], [23], [27] for the planted (l, d)-motif problem. For example, the HAP2 motif "CCAATTA" for the planted $(7, d)$-motif problem is equivalent to the following SPSP representation:

$$P = \begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix}\begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix}$$

and

$$\{s_{i,j}\} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}\begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}\begin{pmatrix} 1 \\ 1 \\ 1 \\ 0 \end{pmatrix}\begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix},$$

with threshold $\alpha = d$.

In order to find the maximum number of binding sites with at most d substitutions from a string motif, the MD problem is to find $\{s_{i,j}\}$ such that for $1 \leq i \leq c = l$, $s_{i,j} = 0$ for a particular j, and $= 1$ for all other j, so as to maximize the number of binding sites as its target function. Note that the SPSP representation P is already fixed, as shown above.
3. Restricting $c = l$, $s_{i,j} = 0$, and $\alpha = 0$, the SPSP representation P is equivalent to a length-l string with wildcard symbols [20], [31]. For example, the BAS2 [37] motif "TAATRA" in string representation with wildcard symbols is equivalent to the following SPSP representation:

$$P = (\mathrm{T})(\mathrm{A})(\mathrm{A})(\mathrm{T})\begin{pmatrix} \mathrm{A} \\ \mathrm{G} \end{pmatrix}(\mathrm{A}) \text{ and}$$

$$\{s_{i,j}\} = (0)(0)(0)(0)\begin{pmatrix} 0 \\ 0 \end{pmatrix}(0),$$

with threshold $\alpha = 0$.

In order to find a set of binding sites with a minimum z-score [31] or p-value [20], the MD problem is to find the SPSP representation P such that for all i, j, $s_{i,j} = 0$, so as to minimize the z-score or p-value of the binding sites as its target function. Note that the z-score or p-value decreases with the inverse of the number of binding sites and the number of conserved symbols.

2.2 Restricted Motif Discovering Problem

In the real biological situation, transcription factors bind to binding sites by some components called DNA-binding domains (for example, zinc finger). Each domain of the transcription factor usually binds to 3-4 bp consecutive regions of the binding sites [24], [35]. Therefore, we may assume the length l_i of each pattern $P_{i,j}$ is not larger than 4. Besides, the background occurrence probability of each length-l pattern in the input sequence is not the same. This uneven probability can be estimated by an order 0 to 3 HMM [36].

Instead of solving the general MD Problem described in Section 2.1, this paper tackles a "restricted" version of the motif problem based on the assumption that l_i is small, that is, $l_i \leq l_{\max}$ for a predefined value $l_{\max}$. Besides, the overall binding site patterns should be similar, that is, the score $s_{i,j}$ of each length-l_i pattern $P_{i,j}$ must be equal to its Hamming distance with some representative length-l_i string R_i:

$$P_i = \begin{pmatrix} \mathrm{ACG} \\ \mathrm{ACT} \\ \mathrm{AGT} \\ \mathrm{CCG} \end{pmatrix} \text{ and } S_{ij} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 2 \end{pmatrix},$$

if $R_i =$ "ACT". Similar if $R_i =$ "ACG", then

$$S_{ij} = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 1 \end{pmatrix}.$$

A length-l string σ is a binding site of M if and only if $\mathrm{score}(\sigma, P) \leq d$, that is, σ should be within Hamming distance d from a particular motif pattern.

Intuitively, the *RMD* Problem is finding an SPSP representation P such that the number of possible string patterns for binding sites $\prod_i |P_i| = w$ is minimized, and at the same time, P can cover the maximum number of binding sites b.

For example, assume the occurrence probability of each length-l pattern is the same. Given the following binding sites $\{s_i\}$ and motif P_1 and P_2:

$$
\begin{array}{cccccccc}
 & & & & & & & \\
s_1 & \mathrm{G} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{T} & \mathrm{A} & \mathrm{A} \\
s_2 & \mathrm{G} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{T} & \mathrm{A} & \mathrm{G} \\
s_3 & \mathrm{G} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{A} & \mathrm{A} & \mathrm{C} \\
s_4 & \mathrm{G} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{A} & \mathrm{A} & \mathrm{G} \\
s_5 & \mathrm{G} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{G} & \mathrm{A} & \mathrm{G} \\
s_6 & \mathrm{G} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{C} & \mathrm{A} & \mathrm{G} \\
s_7 & \mathrm{C} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{G} & \mathrm{A} & \mathrm{C} \\
s_8 & \mathrm{C} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{C} & \mathrm{A} & \mathrm{G} \\
s_9 & \mathrm{C} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{A} & \mathrm{A} & \mathrm{G} \\
s_{10} & \mathrm{C} & \mathrm{T} & \mathrm{A} & \mathrm{T} & \mathrm{G} & \mathrm{A} & \mathrm{C}
\end{array}
$$

$$
P_1 = (\mathrm{GTAT}) \begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \\ \mathrm{T} \end{pmatrix} (\mathrm{A}) \begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \end{pmatrix},
$$

$$
\left\{ s_{i,j} \right\}_1 = (0) \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \end{pmatrix} (0) \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix},
$$

$$
P_2 = \begin{pmatrix} \mathrm{G} \\ \mathrm{C} \end{pmatrix} (\mathrm{TAT}) \begin{pmatrix} \mathrm{A} \\ \mathrm{C} \\ \mathrm{G} \end{pmatrix} (\mathrm{A}) \begin{pmatrix} \mathrm{C} \\ \mathrm{G} \end{pmatrix},
$$

$$
\left\{ s_{i,j} \right\}_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix} (0) \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} (0) \begin{pmatrix} 0 \\ 1 \end{pmatrix}.
$$

Score $\{s_{i,j}\}$ are defined such that $\mathrm{score}(\sigma, P) = \mathrm{Hamming}$ distance between σ and "GTATAAC." Since the number of possible patterns of binding sites for P_1 and P_2 are the same, that is, $w = 4 \times 3$ and $2 \times 3 \times 2$, respectively, P_2 is more likely to be a correct motif than P_1 as P_2 covers more binding sites (s_3 to s_{10}) than P_1 (s_1 to s_6).

Usually, it might not be so obvious which motif is more likely to be correct, for example, when $w_1 < w_2$ and $b_1 < b_2$. In such case, we compare two motifs by the occurrence probabilities (p-values) of their corresponding binding sites in T with the assumption that T is a set of random sequences generated according to a Markov model. Given a motif with $\prod_i |P_i| = w$ (the w possible binding sites patterns are $\{B_k, k = 1, \ldots, w\}$) and having b binding sites in T, the occurrence probability of $\geq b$ binding sites in a set of random sequences can be calculated as

$$
p - \mathrm{value} = 1 - \sum_{i=0}^{b-1} \binom{t(n-l+1)}{i} \left(\sum_{k=1}^{w} P(B_k) \right)^i \left(1 - \sum_{k=1}^{w} P(B_k) \right)^{t(n-l+1)-i},
$$

where $P(B_k)$ is the probability that B_k occurs at a particular position given that the sequence is generated according to the Markov model. A motif with low p-value means that it is likely to be an answer. Note that p-value increases as w and $P(B_k)$, the number and the occurrence probabilities of the possible binding site patterns B_k, increases and decreases as b, the number of binding sites increases. Thus, we define the RMD Problem formally as follows:

RMD Problem. Given the Markov model for the background sequences, t length-n DNA sequences T, the threshold value d, and $l_{\max}$, we want to find a length-l motif P and a set of score $\{s_{i,j}\}$ such that $s_{i,j}$ is equal to the Hamming distance between $P_{i,j}$ and some representative length-l_i string R_i and having the minimum p-value of the corresponding binding sites.

Although we have imposed restriction to the MD problem, this restricted SPSP representation is still more descriptive than the string representation in the sense that all string representations are the special cases of this restricted representation. This restricted representation takes into account the dependence of the occurrence of the nucleotides in a binding site, and some nucleotides in binding sites are conserved. Under the RMD problem, all possible binding sites have equal occurrence probability given that they are generated according to the motif. Note that it is not the same as the occurrence probabilities of the binding sites generated according to the Markov model (background). Thus, we cannot determine whether this restricted representation is more descriptive than the matrix representation. Besides, as we assume that the transcription factor binds to the binding sites by DNA-binding domains at short consecutive regions, the RMD problem does not model binding sites with dependency over a long region, for example, GAL4, functioned as a homodimer with binding pattern $\mathrm{CGGN}_{11}\mathrm{CCG}$ (the prefix CGG is a reverse complement of the suffix CCG), dependency over a long region.

We shall show in Section 3 that there is an efficient heuristic to solve the RMD problem with which we can successfully find motifs in some cases for which popular motif-finding software fail.

3 Algorithm SPSP Finder

In this section, we describe a heuristic algorithm, *SPSP Finder*, to solve the RMD Problem. This algorithm starts with a set of "good" string patterns and, based on local search, finds some local optimal SPSP representations and their corresponding binding sites. This algorithm has two main steps. The first step, served as seed searching, is to find a set of length-l string motifs with many binding sites in the input sequences. In the second step, we start with each length-l string R as a seed SPSP representation and merge some positions of R's binding sites to form another SPSP representation with smaller p-value. This merging step is repeated until the p-value cannot be further reduced. Definitely, this algorithm cannot guarantee the finding optimal motif in SPSP representation. However, when more seed sequences are considered, the longer is the running time, the better will be the solution.

3.1 Seed Searching

Voting Algorithm [9], [17] is applied to discover length-l string motifs. Voting is used for solving the planted (l, d)-motif problem where a motif is represented by a length-l string S, and the binding sites are d-variants of S (*d-variant* of S is a length-l string derivable from S with at most d symbol substitutions). This algorithm is based on the idea that if each length-l substring in the input sequences T gives a vote to each of its d-variants, a string S with b d-variants in T will get exactly b votes. Finding the number of d-variants in T of each length-l string S takes $O(nt(3l)^d)$ time [9], [17].

Since the occurrence probability of each length-l string in T is different, we modify the Voting Algorithm such that each string σ gives $1/P(\sigma)$ vote to each of its d-variants, where $P(\sigma)$ is σ's occurrence probability based on the background modeled by HMM. A string σ with low occurrence probability in the background will contribute a higher score to its d-variants. As the string S with relatively more d-variants of low occurrence probabilities is more

likely to be one of the motif patterns, we refine each length-l string one by one (Section 3.2) in decreasing order of the sum of (weighted) votes received.

3.2 Refining the SPSP Representation

Given representative string S (motif candidate), we can find all length-l d-variants of S (potential binding sites) in the t length-n DNA sequences T. By aligning these length-l d-variants, we can construct a restricted SPSP representation P for these d-variants by considering the consensus substrings as the representative strings R_i. However, as some of these d-variants might not be binding sites, the value of $\sum_k P(k)$, as well as the p-value may be very large. In order to reduce the value of $\sum_k P(k)$, we shall construct restricted SPSP representation for subsets of the d-variants. Since finding the optimal subset of d-variants (the subset with the lowest p-value) is NP-complete (see Appendix), heuristic approach is being considered. We begin with the set of all d-variants. At each iteration, we remove the d-variant whose removal decreases the p-value most. Note that the restricted SPSP representation will change if the set of d-variants is different. If we find a motif candidate M with smaller p-value than the best motif M^* found so far, we update M^* by M. We repeat this step until the p-value of a new motif candidate M cannot be lowered.

After considering (or refining) one string, we shall consider (or refine) the next candidate string having the largest weighted votes. When the number of d-variants of the remaining candidates (as the candidates have been sorted in decreasing order of weighted votes) is too small to be refined to a better motif than M^*, we stop the process and report M^* as the answer.

4 Experimental Results

Based on the ideas in Section 3, we have implemented SPSP Finder in C++. SPSP Finder was used to find motifs in both simulated and real biological data. All experiments were performed on a 2.4-GHz P4 CPU with 1 Gbyte of memory. The performance of SPSP Finder was compared with various existing motif-finding algorithms.

4.1 Simulated Data

The simulated data were generated in the following manner. Twenty length-600 sequences were generated with each nucleotide having the same occurrence probability 0.25. Then, a length-l motif M in SPSP representation with $l_{\max} = 4$ was picked randomly according to the following steps:

1. A set of c numbers $l_1 \ldots l_c$ such that $c \leq l$ and $\sum_i l_i = l$, corresponding to the parameters of an SPSP representation, was generated randomly.
2. For $i = 1, \ldots, c$, an integer r_i was randomly picked from 1 to 4 with equal probability. r_i length-l_i random strings were generated independently with each nucleotide having the same occurrence probability 0.25.

A binding site of M was randomly picked with equal probability and planted at a random position of each sequence in T. The Weeder [25], MEME [16], AlignACE [12], and SPSP Finder were used to discover this hidden motif M. SPSP Finder calculated the score of a predicted motif using an order-0 Markov model with the same occurrence probability for each nucleotide. The accuracy for each motif predicted by the above algorithms is defined as

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

A planted binding site is *correctly predicted* if that binding site overlaps with at least one predicted binding site. An algorithm is said to have predicted the hidden motif correctly if the accuracy $\geq$ 0.5. For each set of parameters, that is, length l and threshold (Hamming distance) α, we ran 50 test cases. Table 1 shows the success rate of the algorithms in discovering the motif.

TABLE 1
Experimental Results on Simulated Data—The Success Rate of Each Software

l	α	Weeder	MEME	AlignACE	SPSP-Finder
5	0	100%	68%	10%	100%
5	1	6%	16%	4%	42%
7	1	100%	72%	14%	100%
7	2	7%	12%	2%	38%
9	2	100%	68%	18%	100%
9	3	2%	12%	8%	52%
11	3	98%	74%	14%	100%
11	4	3%	14%	0%	32%

Buhler and Tompa [4] proved that when α is large with respect to l (for example, the (5,1), (7,2), (9,3), and (11,4) problems), there are many random patterns having the same number of α-variants as the motif; so, algorithms are unlikely to be able to discover the motif without extra information. Indeed, the Weeder, MEME, and AlignACE do not perform well in these cases. MEME has a better performance than the other two algorithms because it allows different occurrence probabilities for nucleotides at each position. Since SPSP Finder considers the dependence of the nucleotides, it has better performance than Weeder, MEME, and AlignACE.

4.2 Real Biological Data

SCPD [37] contains information of different transcription factors for yeast. For each set of genes regulated by the same transcription factor, we chose the 600 base pairs in the upstream of these genes as the input sequences T. The Weeder, MEME, AlignACE, and SPSP Finder were used to discover the motifs. SPSP Finder used an order-0 Markov model calculated based on the input sequence when calculating score of each predicted motif. Table 2 showed the experimental results of all transcription factors in SCPD except those motifs, which cannot be discovered by any algorithms. As shown in Table 2, SPSP Finder performs better than other algorithms in most cases. There are six motifs, ACE2, AP1, BAS2, CSRE, HAP2/3/4, and UASCAR, and their binding sites could be discovered (accuracy $\geq$ 0.5) by SPSP Finder but not by the other algorithms. Refer to the published binding sites in SCPD database where there are nucleotide dependencies in these binding sites.

For example, the HAP2/3/4 complex is a CCAAT-binding complex, which mainly binds to the sequence

TABLE 2
Experimental Results on Real Biological Data in SCPD Database

Factor Name	Pattern	Weeder	MEME	AlignACE	SPSP-Finder
13nt	ACGAGGCTTACCG	-	-	ACGAGGCTTACCG	(ACGA)(GGCT)(TACC)(G)
ACE2	GCTGGT	-	-	-	(GCTG)(GT)
ADR1	TCTCC	-	TCTCC	TCTCC	(TCTC)(C)
AP1	TTANTAA	-	-	-	(TTA)$\binom{C}{G}$(TAA)
BAS2	TAATGA	-	-	-	(TAAA)$\binom{A}{G}$(A)
CCBF	CNCGAAA	CACGAAA	-	-	(C)$\binom{A}{C}{G}{T}$(CGAA)(A)
CPF1	TCACGTG	CACGTG	TCACGTG	-	(CACG)(TG)
CSRE	CGGAYRRAWGG	-	-	-	(CGGA)$\binom{TGA}{TAA}{CGG}$(A)$\binom{A}{T}$(GG)
CuRE	TTTGCTC	TTTGCTCA		-	(TTT)$\binom{GCT}{TCG}$(C)
GATA	CTTATC	CTTATC	-	-	(CTTA)(TC)
HAP2/3/4	CCAATCA	-	-	-	(CCAA)$\binom{TC}{TG}{CC}$(A)
LEU	CCGNNNNCGG	CCGGGACCGG	CCGGAACCGG	-	(CGG)$\binom{A}{G}$(ACCGG)
MATα2	CRTGTWWWW	CATGTAATTA	-	CATGTAATT	$\binom{GA}{GT}{TA}$(AATT)$\binom{AC}{TA}{TC}{TG}\binom{A}{C}$
MCM1	CCNNNWWRGG	CCCGTTTAGG	CCTAATTAGG	-	-
SFF	GTMAACAA	-	GTCAACAA	-	-
UASCAR	TTTCCATTAGG	-	-	-	(T)$\binom{GCCC}{TCAC}{TCCA}$(TT)$\binom{AGCG}{AGGA}$

Motifs of transcription factors that cannot be found by any algorithms were not shown in this table. 'M' stands for 'A' or 'C', 'N' stands for any nucleotide. 'R' stands for 'A' or 'G', 'W' stands for 'A' or 'T', 'Y' stands for 'C' or 'T'. Those motifs that all four algorithms can/cannot discover are not shown.

"CCAATCA" in yeast. Although it also binds to the sequences "CCAATGA" and "CCAACCA," it cannot bind to sequences "CCAAACA" and "CCAAAGA" [30]. Since the binding sites are short and there are two nonconserved positions (positions 5 and 6), Weeder failed to discover the published motif because there were many length-7 random patterns whose 2-variants occurred more frequently than the binding sites of "CCAATCA." In this case, Weeder cannot distinguish the published motif "CCAATCA" from these random patterns. Similarly, MEME and AlignACE failed because there were many PSSMs having higher scores than the score of the published motif if the nucleotide dependency in positions 5 and 6 were not considered. By considering the nucleotide dependency in positions 5 and 6, the SPSP Finder discovered the motif in the SPSP representation:

$$(\mathrm{CCAA})\begin{pmatrix}\mathrm{TC}\\ \mathrm{TG}\\ \mathrm{CC}\end{pmatrix}(\mathrm{A}),$$

which had a lower p-value than "CCAAYSA" and other random patterns.

CSRE is a transcription factor responsible for the transcriptional activation of gluconeogenic structural genes. There are five binding sites in the data set that can be represented by the motif "CGGAYRRAWGG." This motif contains four wildcard symbols and represents 16 different binding sequences instead of 5. Since this motif cannot model the binding sites specifically, many length-11 random patterns had frequently occurring 4-variants and could be mistaken as the hidden motif. Therefore, Weeder could not discover the motif. Similarly, MEME and AlignACE failed even using the more precise PSSM representation. SPSP Finder discovered the following motif in SPSP representation:

$$(\mathrm{CGGA})\begin{pmatrix}\mathrm{TGA}\\ \mathrm{TAA}\\ \mathrm{CGG}\end{pmatrix}(\mathrm{A})\begin{pmatrix}\mathrm{A}\\ \mathrm{T}\end{pmatrix}(\mathrm{GG}).$$

Although this motif in SPSP representation represented six instead of five binding patterns, it could describe the binding sites better than those motifs in string representation or PSSM. Therefore, SPSP Finder could discover the published motif successfully, whereas Weeder, MEME, and AlignACE failed.

For those cases that SPSP Finder and other algorithms could discover the published binding sites, SPSP Finder had an advantage that it can represent the binding sites better. For example, the CCBF transcription factor can bind to sequences "CNCGAAA," where "N" represents any nucleotides. Although both Weeder and SPSP Finder could discover the published motif, Weeder represented the motif as "CACGAAA" with at most 1 point substitution that will wrongly consider "TACGAAA," "CAAGAAA," etc., as binding sites. On the other hand, SPSP Finder represented the motif in the following format:

TABLE 3
Experimental Results on Real Biological Data in TRANSFAC Database

Factor Name	Pattern	Weeder	MEME	AlignACE	SPSP-Finder
Ac	CGCAGGTG	CGCAGGTG	CGCAGGTG	-	(CGCA)[GGTG GCTC]
Antp	TTWTYMT	-	ATTTTTA	-	-
AS-CT3	CAGGTG	CAGGTG	-	-	(CAGG)(TG)
BEAF-32B	CGATA	-	-	-	(CGAT)(A)
Cad	TTTAKG	-	-	-	(TTTA)(GG)
Ci	TGGGTGGTC	GGGTGGTCCA	GGGTGGACC	GGGTGGTCC	[TTT GATG](GGGT)(GG)
D_MEF2	TTAAAAATAA	-	-	-	(TTTT)[AA CG](AAA)[T A]
D1	GGGTTTTTCCN	-	GGTTTTTCCCA	-	-
DREF	ASCTATC GATADNY	GCCACC TATCGA	GCCACCT ATCGATA	-	(AGC)[TA TT](TCG)[ATA AAT TTT](A)
Eve	TNWSSYCTGC	-	-	-	[TTA TTC TTG GTG][GCT GCC GCA](CTCC)
GCM	NNACCCGCATNNN	ACCCTCATGAGT	-	-	-
Kr	AMYGGGTTAW	-	-	ACGGGTTAAGC	[TAAA TCGA GGGT](AGGG)[TT AT]
Sc	CGCAGGTG	CGCAGGTG	CGCAGGTG	-	(CGCA)[GGTG GCTC]
Su_Hw	YRYTGCATAYYY	-	-	-	(T)[G A](TTGC)(ATAC)
TBP	STATAAAW	-	-	-	[GCT ACC CCC](ATAA)(A)
Zeste	WNTTGAGTGNN	-	ACTTGAGTGAG	TTTGAGTGAGT	[TT TC GT](GAGT)(GTTT)(T)

Motifs of transcription factors that cannot be found by any algorithms were not shown in this table. 'M' stands for 'A' or 'C', 'N' stands for any nucleotide. 'D' stands for 'A', 'G' or 'T', 'K' stands for 'G' or 'T', 'R' stands for 'A' or 'G', 'S' stands for 'C' or 'G', 'W' stands for 'A' or 'T', 'Y' stands for 'C' or 'T'. Those motifs that all four algorithms can/cannot discover are not shown.

$$(\mathrm{C})\begin{pmatrix}\mathrm{A}\\ \mathrm{C}\\ \mathrm{G}\\ \mathrm{T}\end{pmatrix}(\mathrm{CGAA})(\mathrm{A}),$$

which can represent the motif better than Weeder. Similarily, SPSP Finder had better representations for the CuRE, LEU, and MATα2 motifs.

There were two cases that SPSP Finder failed while some of the other algorithms were successful. The SPSP Finder could not discover the motifs of MCM1 (SPSP Finder discovered the published motif at rank 25), whereas Weeder was successful because there were no strong bias at most positions of this motif, and the information contained in the input sequences was little. Weeder could discover the motifs because it had extra information about different background models for discovering motifs in different species. Since SPSP Finder only constructed a Markov model from the input sequences, it did not have any extra information on the background model and thus failed to discover the motif.

Similarly, the SPSP Finder could not discover the motifs of MCM1 and SFF, whereas MEME were successful because there were no strong bias at most positions of this motif. In these cases, a matrix representation can model the motif better than a string representation, and the restricted SPSP representation used in RMD problem (because PSSM or PWM is a more direct and efficient representations in these cases). Excluding these two data sets, SPSP Finder had the best performance among the algorithms.

We have also tested the performance of SPSP Finder on the fruit-fly data from the TRANSFAC database [38]. Experimental results were shown in Table 3. SPSP Finder also had the best performance among the four algorithms. Among the 16 data sets, a total of six motifs, BEAF-32B, Cad, D_MEF2, Eve, Su_Hw, and TBP, and their binding sites could be discovered by SPSP Finder but not by the other algorithms. Again, we did not list out those motifs that could not be discovered by any of the algorithms.

5 Concluding Remarks

In this paper, we have proposed a new and better representation based on SPSP to describe a motif and its binding sites. With the proposed heuristic algorithm for the RMD Problem, we can successfully find motifs and their binding sites even in some situations in which existing popular software fail. In the RMD problem, the possible scores received by the binding sites are limited to a small set of integers. In the real biological situation, each binding site should have a different score. With this assumption, we would expect an increase in the success rate of finding the correct motif. However, finding the optimal motif for the general MD Problem without restrictions is very difficult and should be no easier than finding the optimal motif in matrix representation. The difficulty lies not only with the large solution space of the score $\{s_{i,j}\}$, but also with the exponential number of possible sets of patterns for a length-l

motif. At this moment, no heuristic algorithm for the general MD problem with reasonable performance is known.

APPENDIX

RESTRICTED MOTIF DISCOVERING PROBLEM IS NP-COMPLETE

In this section, we will show that the Restricted Motif Discovering (RMD) Problem is NP-complete. In order to answer whether the RMD problem is NP-complete, we convert the RMD problem into a decision problem:

RMD Decision Problem. *Given t length-n DNA sequences T and we assume the occurrence probability of each length-l pattern in the input sequences T is the same, whether there exists a motif P, $\prod_i |P_i| \leq w$ (that is, $\sum P(B_k) = w/4^l$), that has exactly b binding sites in T?*

It is easy to see that the RMD decision problem is in NP because given a motif P, we can verify whether P has b binding sites in T and $\prod_i |P_i| \leq w$ in polynomial time. In order to show that the RMD decision problem is NP-complete, we reduce the *Clique Decision Problem* (CDP) to it.

Clique Decision Problem (CDP). *Given a graph $G = (V, E)$ and an integer $k > 0$, the CDP is to determine whether G contains a clique of size k.*

Denote $V = \{v_i, 1 \leq i \leq n\}$ and $E = \{e_j, 1 \leq j \leq m\}$. Let $\deg(v_i)$ be the degree of vertex v_i and $D = \max_i\{\deg(v_i)\}$. We construct $2n$ length-$(nD - m)$ DNA sequences as follows: For each vertex v_i, a length-$(nD - m)$ DNA sequence σ_i representing a binding site is constructed such that σ_i has all symbols "T" except D symbols of "A" or "C." The first m symbols of these n sequences (one for each vertex) resemble the incidence matrix such that the jth symbols of σ_i and $\sigma_{i'}$ are "A" and "C" corresponding to the jth edge connecting v_i and $v_{i'}$, respectively. Thus, σ_i should have $\deg(\sigma_i)$ symbols of "A" or "C" in its first m symbols. If $\deg(\sigma_i) < D$, then $D - \deg(\sigma_i)$ symbols of "A" will be packed after the first m symbols such that no two sequences have symbol "A" at the same position and each σ_i has exactly D symbols of "A" or "C." Precisely, we have

$$\sigma_i[j] = \begin{cases} \text{'A'} & \exists \mu, \{v_\mu, v_i\} = e_j, \mu > i \\ \text{'C'} & \exists \mu, \{v_\mu, v_i\} = e_j, \mu < i \\ \text{'A'} & |E| + \sum_{i'=1}^{i-1}(D - \deg(v_{i'})) < j \leq |E| \\ & \quad + \sum_{i'=1}^{i}(D - \deg(v_{i'})) \\ \text{'T'} & \text{otherwise.} \end{cases}$$

Denote this set of n strings by T_1.

In addition to these n length-$(nD - m)$ DNA sequences, we have another n length-$(nD - m)$ DNA sequences with symbol "T" only. Denote this set of strings by T_2, $T = T_1 \cup T_2$. We solve the RMD decision problem with $l = \alpha = nD - m$, $l_{\max} = 1$, and $w = 2^{kD-k(k-1)}3^{k(k-1)/2}$. If there exists a motif P, $\prod_i |P_i| \leq w$ having $b = n + k$ binding sites in T, the answer of CDP is "yes," otherwise, the answer is "no." Fig. 1 shows an example of this reduction. Theorem 1 proves the correctness of this reduction.

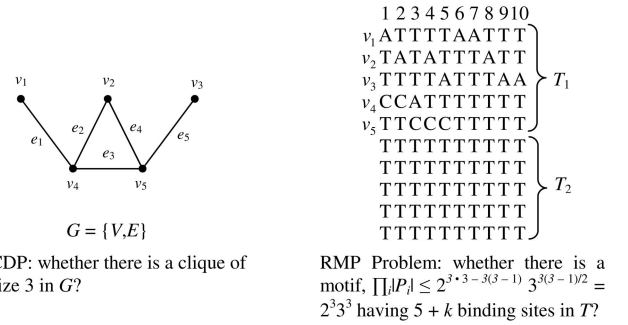

Fig. 1. An example of the reduction from CDP to RMD decision problem.

Theorem 1. *There is a motif P with $\prod_i |P_i| \leq 2^{kD-k(k-1)}3^{k(k-1)/2}$ and having $b = n + k$ binding sites in T if and only if there is a clique of size k in G.*

Proof. Without loss of generality, assume that $\{v_i \mid 1 \leq i \leq k\}$ with $\{e_j \mid 1 \leq j \leq k(k-1)/2\}$ forms a clique of size k in G, the set of binding sites should contain all the strings in T_2 and k strings (corresponding to the vertices of the clique) in T_1, that is, $n + k$ strings. The motif should have its first $k(k-1)/2$ positions having the symbols "A," "C," and "T," that is,

$$P_1 = P_2 = \ldots = P_{k(k-1)/2} = \begin{pmatrix} \text{A} \\ \text{C} \\ \text{T} \end{pmatrix}$$

and exactly $kD - k(k-1)$ positions having the pair of symbols "A," "T" or "C," "T." Note that all the other positions should be conserved and have the symbol "T." Thus, these $n + k$ strings can be represented in SPSP representation with $\prod_i |P_i| = 2^{kD-k(k-1)}3^{k(k-1)/2}$.

Assume there is a motif P in the SPSP representation with $\prod_i |P_i| \leq 2^{kD-k(k-1)}3^{k(k-1)/2}$ and having exactly $n + k$ binding sites and y $(y \geq k)$ out of these $n + k$ binding sites in set T_1. Since each binding site in T_1 has exactly D symbols of "A" or "C" and each of these yD symbols can be either represented by a partition with two symbols or three symbols, we have

$$\begin{pmatrix} \text{A} \\ \text{T} \end{pmatrix} \text{ or } \begin{pmatrix} \text{C} \\ \text{T} \end{pmatrix} \text{ or } \begin{pmatrix} \text{A} \\ \text{C} \\ \text{T} \end{pmatrix}.$$

The motif P has the smallest $\prod_i |P_i|$ when it has the largest possible number $y(y-1)/2$ of partitions with three symbols and the smallest value of y $(y = k)$. We have $\prod_i |P_i| \geq 2^{yD-y(y-1)}3^{y(y-1)/2} \geq 2^{kD-k(k-1)}3^{k(k-1)/2}$. Therefore, $\prod_i |P_i| = 2^{kD-k(k-1)}3^{k(k-1)/2}$, and $y = k$. Since 2 and 3 are prime numbers, there are $k(k-1)/2$ pattern sets P_i with three symbols of "A," "C," and "T," and the corresponding vertices of the k sequences in T_1 form a clique of size k in G. □

ACKNOWLEDGMENTS

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

REFERENCES

[1] T. Bailey and C. Elkan, "Unsupervised Learning of Multiple Motifs in Biopolymers Using Expectation Maximization," *Machine Learning,* vol. 21, pp. 51-80, 1995.

[2] Y. Barash, G. Bejerano, and N. Friedman, "A Simple Hyper-Geometric Approach for Discovering Putative Transcription Factor Binding Sites," *Proc. Int'l Workshop Algorithms in Bioinformatics (WABI '01),* pp. 278-293, 2001.

[3] Y. Barash, G. Elidan, N. Friedman, and T. Kaplan, "Modeling Dependencies in Protein-DNA Binding Sites," *Proc. Ann. Int'l Conf. Research in Computational Molecular Biology (RECOMB '03),* pp. 28-37, 2003.

[4] J. Buhler and M. Tompa, "Finding Motifs Using Random Projections," *Proc. Ann. Int'l Conf. Research in Computational Molecular Biology (RECOMB '01),* pp. 69-76, 2001.

[5] 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," *Nucleic Acids Research,* vol. 30, pp. 1255-1261, 2002.

[6] F. Chin and H. Leung, "An Efficient Algorithm for String Motif Discovery," *Proc. Asia-Pacific Bioinformatics Conf. (APBC '06),* pp. 79-88, 2006.

[7] F. Chin and H. Leung, "An Efficient Algorithm for the Extended (l, d)-Motif Problem with Unknown Number of Binding Sites," *Proc. Int'l Symp. BioInformatics and BioEngineering (BIBE '05),* pp. 11-18, 2005.

[8] F. Chin and H. Leung, "Voting Algorithms for Discovering Long Motifs," *Proc. Asia-Pacific Bioinformatics Conf. (APBC '05),* pp. 261-271, 2005.

[9] 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," *Proc. Ann. Int'l Conf. Research in Computational Molecular Biology (RECOMB '04),* pp. 125-132, 2004.

[10] S. Hannenhalli and L.S. Wang, "Enhanced Position Weight Matrices Using Mixture Models," *Bioinformatics,* vol. 21, pp. 204-212, 2005.

[11] 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," *Proc. Third Int'l Conf. Bioinformatics and Genome Research,* pp. 201-216, 1995.

[12] 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*," *J. Molecular Biology,* vol. 296, no. 5, pp. 1205-1214, 2000.

[13] U. Keich and P. Pevzner, "Finding Motifs in the Twilight Zone," *Proc. Ann. Int'l Conf. Research in Computational Molecular Biology (RECOMB '02),* pp. 195-204, 2002.

[14] S. Kielbasa, J. Korbel, D. Beule, J. Schuchhardt, and H. Herzel, "Combining Frequency and Positional Information to Predict Transcription Factor Binding Sites," *Bioinformatics,* vol. 17, pp. 1019-1026, 2001.

[15] C. Lawrence, S. Altschul, M. Boguski, J. Liu, A. Neuwald, and J. Wootton, "Detecting Subtule Sequence Signals: A Gibbs Sampling Strategy for Multiple Alignment," *Science,* vol. 262, pp. 208-214, 1993.

[16] 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,* vol. 7, pp. 41-51, 1990.

[17] H. Leung and F. Chin, "Algorithms for Challenging Motif Problems," *J. Bioinformatics and Computational Biology,* pp. 43-58, 2005.

[18] H. Leung and F. Chin, "Discovering Motifs with Transcription Factor Domain Knowledge," *Proc. Pacific Symp. Biocomputing (PSB '07),* pp. 472-483, 2007.

[19] H. Leung and F. Chin, "Finding Exact Optimal Motif in Matrix Representation by Partitioning," *Bioinformatics,* vol. 22, pp. 86-92, 2005.

[20] H. Leung and F. Chin, "Generalized Planted (l, d)-Motif Problem with Negative Set," *Proc. Int'l Workshop Algorithms in Bioinformatics (WABI '05),* pp. 264-275, 2005.

[21] H. Leung, F. Chin, S.M. Yiu, R. Rosenfeld, and W.W. Tsang, "Finding Motifs with Insufficient Number of Strong Binding Sites," *J. Computational Biology,* vol. 12, no. 6, pp. 686-701, 2005.

[22] M. Li, B. Ma, and L. Wang, "Finding Similar Regions in Many Strings," *J. Computer and System Sciences,* vol. 65, pp. 73-96, 2002.

[23] S. Liang, "cWINNOWER Algorithm for Finding Fuzzy DNA Motifs," *Proc. IEEE CS Bioinformatics Conf.,* pp. 260-265, 2003.

[24] 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," *Nucleic Acids Research,* vol. 29, pp. 2471-2478, 2001.

[25] G. Pavesi, P. Mereghetti, F. Zambelli, M. Stefani, G. Mauri, and G. Pesole, "MoD Tools: Regulatory Motif Discovery in Nucleotide Sequences from Co-Regulated or Homologous Genes," *Nucleic Acids Research,* vol. 34, pp. 566-570, 2006.

[26] G. Pesole, N. Prunella, S. Liuni, M. Attimonelli, and C. Saccone, "Wordup: An Efficient Algorithm for Discovering Statistically Significant Patterns in DNA Sequences," *Nucleic Acids Research,* vol. 20, no. 11, pp. 2871-2875, 1992.

[27] P. Pevzner and S.H. Sze, "Combinatorial Approaches to Finding Subtle Signals in DNA Sequences," *Proc. Eighth Int'l Conf. Intelligent Systems for Molecular Biology,* pp. 269-278, 2000.

[28] S. Rajasekaran, S. Balla, and C.H. Huang, "Exact Algorithms for Planted Motif Challenge Problem," *Proc. Asia-Pacific Bioinformatics Conf. (APBC '05),* pp. 249-259, 2005.

[29] S. Sinha, "Discriminative Motifs," *Proc. Sixth Ann. Int'l Conf. Computational Biology,* pp. 291-298, 2002.

[30] S. Sinha, S.N. Maity, J. Lu, and B. Crombrugghe, "Recombinant Rat CBF-C, the Third Subunit of CBF/NFY, Allows Formation of a Protein-DNA Complex with CBF-A and CBF-B and with Yeast HAP2 and HAP3," *Proc. Nat'l Academy of Sciences,* vol. 92, no. 5, pp. 1624-1628, 1995.

[31] S. Sinha and M. Tompa, "A Statistical Method for Finding Transcription Factor Binding Sites," *Proc. Eighth Int'l Conf. Intelligent Systems for Molecular Biology,* pp. 344-354, 2000.

[32] K.T. Takusagawa and D.K. Gifford, "Negative Information for Motif Discovery," *Proc. Pacific Symp. Biocomputing (PSB '04),* pp. 360-371, 2004.

[33] M. Tompa, "An Exact Method for Finding Short Motifs in Sequences with Application to the Ribosome Binding Site Problem," *Proc. Seventh Int'l Conf. Intelligent Systems for Molecular Biology,* pp. 262-271, 1999.

[34] Y. Xing, J.D. Fikes, and L. Guarente, "Mutations in Yeast HAP2 HAP3 Define a Hybrid CCAAT Box Binding Domain," *EMBO J.,* vol. 12, pp. 4647-4655, 1993.

[35] S. Wolfe, H. Greisman, E. Ramm, and C. Pabo, "Analysis of Zinc Fingers Optimized via Phage Display: Evaluating the Utility of a Recognition Code," *J. Molecular Biology,* vol. 285, no. 5, pp. 1917-1934, 1999.

[36] X. Zhao, H. Huang, and T.P. Speed, "Finding Short DNA Motifs Using Permuted Markov Models," *Proc. Ann. Int'l Conf. Research in Computational Molecular Biology (RECOMB '04),* pp. 68-75, 2004.

[37] J. Zhu and M. Zhang, "SCPD: A Promoter Database of the Yeast *Saccharomyces cerevisiae*," *Bioinformatics,* vol. 15, pp. 563-577, http://cgsigma.cshl.org/jian/, 1999.

[38] TRANSFAC Database, http://www.gene-regulation.com/pub/databases.html, 2007.

Francis Chin received the BASc degree from the University of Toronto, Canada, in 1972, and the MS, MA, and PhD degrees from Princeton University, in 1974, 1975, and 1976, respectively. Since 1975, he has taught at the University of Maryland, Baltimore County, the University of California, San Diego, the University of Alberta, Chinese University of Hong Kong, and the University of Texas, Dallas. He joined the University of Hong Kong (HKU) in 1985, where he was the founding head of the department from its establishment until 31 December 1999 and is currently serving as the associated dean of engineering and the chair of the Department of Computer Science. Between 1992-1996, he served as the associated dean of the Graduate School. His research interests include bioinformatics, design and analysis of algorithms, and online algorithms. He is currently serving as a manager editor of the *International Journal on Foundations of Computer Science* and is also on the editorial boards of several journals. He has served on the program committees and as conference chairman of numerous international workshops and conferences. In 1996, he was elected as an IEEE Fellow.

Henry C.M. Leung received the PhD degree from the University of Hong Kong, where he worked on the motif discovering for DNA sequences. His research interest is on the motif-discovering problem in DNA sequences and protein sequences. The results have been published in various international conference and journals.