BIOINFORMATICS ORIGINAL PAPER Vol. 25 no. 7 2009, pages 868–874
doi:10.1093/bioinformatics/btp090

Genome analysis

M are better than one: an ensemble-based motif finder and its application to regulatory element prediction

Chen Yanover[1], Mona Singh[2] and Elena Zaslavsky[2,*]

[1]Program in Computational Biology, Fred Hutchinson Cancer Research Center, Seattle, WA and [2]Department of Computer Science and Lewis-Sigler Institute for Integrative Genomics, Princeton University, Princeton, NJ, USA

Received on August 20, 2008; revised on January 7, 2009; accepted on February 12, 2009

Advance Access publication February 17, 2009

Associate Editor: Alfonso Valencia

ABSTRACT

Motivation: Identifying regulatory elements in genomic sequences is a key component in understanding the control of gene expression. Computationally, this problem is often addressed by motif discovery, where the goal is to find a set of mutually similar subsequences within a collection of input sequences. Though motif discovery is widely studied and many approaches to it have been suggested, it remains a challenging and as yet unresolved problem.

Results: We introduce SAMF (Solution-Aggregating Motif Finder), a novel approach for motif discovery. SAMF is based on a Markov Random Field formulation, and its key idea is to uncover and aggregate multiple statistically significant solutions to the given motif finding problem. In contrast to many earlier methods, SAMF does not require prior estimates on the number of motif instances present in the data, is not limited by motif length, and allows motifs to overlap. Though SAMF is broadly applicable, these features make it particularly well suited for addressing the challenges of prokaryotic regulatory element detection. We test SAMF's ability to find transcription factor binding sites in an *Escherichia coli* dataset and show that it outperforms previous methods. Additionally, we uncover a number of previously unidentified binding sites in this data, and provide evidence that they correspond to actual regulatory elements.

Contact: cyanover@fhcrc.org, {msingh,elenaz}@cs.princeton.edu

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

A central challenge in molecular biology is understanding the logic and mechanisms of gene regulation. An important step towards this lies in identifying regulatory elements within genomes. The prominent role of sequence specificity in controlling gene expression allows regulatory element detection to be performed via motif discovery, where the goal is to find approximately repeated patterns in unaligned sequences that are thought to share a common regulator and thus possess a common motif. Such sets of sequences can be obtained through DNA microarray studies (Tavazoie *et al.*, 1999), ChIP-chip (Lee *et al.*, 2002) or ChIP-seq (Robertson *et al.*, 2007) experiments or protein binding microarrays (Mukherjee *et al.*, 2004). Alternatively, sets of regulatory regions of orthologous genes, which may be partly or wholly aligned, can be searched for regulatory elements (e.g. Stark *et al.*, 2007).

Because it represents the most basic type of first step analysis towards uncovering regulatory networks, *de novo* motif finding is a classic and widely studied problem in computational biology. Despite numerous motif finding approaches, based on various formulations and algorithmic techniques [see reviews of Das and Dai (2007); MacIsaac and Fraenkel (2006), and references therein], no single currently existing method can solve them completely [e.g. see Das and Dai (2007); Hu *et al.* (2005); Tompa *et al.* (2005)].

Here, we introduce a novel motif finding algorithm, *SAMF* (Solution-Aggregating Motif Finder), which builds upon powerful techniques in machine learning by modeling the problem as a Markov Random Field (MRF) and iteratively inferring an ensemble of highly probable model configurations (or top scoring solutions) using the Best Max-Marginal First (BMMF) algorithm (Yanover and Weiss, 2004). We utilize an exact calculation of statistical significance (Zaslavsky and Singh, 2006) to determine the number of configurations to be considered, and derive coherent motifs by applying a new technique for aggregating and clustering the ensemble of significant configurations. Each such configuration is assigned a weight that depends on its score in the model, and the algorithm predicts a final set of putative motifs corresponding to sufficiently highly weighted clusters. This ensemble-based procedure enables SAMF to detect both distinct multiple motifs and repeated motif instances within each sequence without requiring an estimate on the number of binding sites. The spirit of our work agrees with the recent observation that considering an ensemble of solutions to motif finding (Reddy *et al.*, 2007) and optimization problems in general (Webb-Robertson *et al.*, 2008) is more suitable than simply searching for a single optimal solution. Our approach and the meaning ascribed to the term 'ensemble' is different from that of a number of recent methods, which combine the findings of several motif discovery algorithms, each potentially utilizing entirely different methodologies and objective functions, to produce a final motif prediction (Hu *et al.*, 2006; Wijaya *et al.*, 2008). Here, we explicitly aim to enumerate and aggregate an ensemble of *distinct* solutions obtained via the same underlying MRF model.

While our method can be used in any motif finding application, whether in DNA, RNA or protein sequences, we apply it to detect regulatory elements in prokaryotic genomes. In prokaryotes, regulation of gene expression is carried out primarily during transcription, and is modulated by transcription factors that carry

*To whom correspondence should be addressed.

out their function by binding DNA fragments in the immediate upstream vicinity of the gene being regulated. Though much recent attention has focused on computational methods for the difficult problem of identifying regulatory elements in eukaryotes [e.g. Tompa *et al.* (2005)], prokaryotic transcription factor binding site detection comes with its own set of challenges. Prokaryotic promoter regions contain binding sites that tend to be long [10–48 bp, Robison *et al.*]; this limits the applicability of motif finders based on pattern enumeration—which are the basis of some of the most successful motif finders in eukaryotes (Elemento *et al.*, 2007; Tompa *et al.*, 2005)—as they typically consider binding sites of at most 12 bp. Additionally, prokaryotic binding sites can overlap and often appear in tandem (Hermsen *et al.*, 2006; Karp *et al.*, 2007), mechanisms that are used to selectively modulate gene expression. The overlapping nature of the binding sites poses a problem to some motif finders (Bailey and Elkan, 1995; Roth *et al.*, 1998), as they are often constrained to look for non-overlapping motifs. Additionally, many methods require the expected number of motifs and their occurrences to be specified as parameters.

Our algorithm, SAMF, is based on a novel formulation and robustly addresses the issues above. The motifs we are able to find are not length constrained and can overlap. We can uncover multiple distinct motifs and multiple occurrences without having to provide an estimate on the number of binding sites. Indeed, we obtain excellent results in practice. In particular, we test SAMF on sets of genes experimentally determined to be regulated by a common *Escherichia coli* transcription factor (McGuire *et al.*, 2000; Robison *et al.*, 1998), and apply our algorithm to look for regulatory motifs in the corresponding upstream regions. We compare our results with those of a representative set of previous approaches, and demonstrate that our algorithm outperforms the others both in sensitivity and specificity when identifying regulatory elements in bacteria. Additionally, we predict a number of putative sites and wholly conserved motifs in this dataset, and provide biological evidence that they likely are real transcription factor binding sites.

2 METHODS

We first introduce a simple motif finding model, one that allows a single motif instance per sequence, and then extend it to multiple motifs.

2.1 Problem formulation

We cast motif discovery as the problem of finding an ungapped local multiple sequence alignment (MSA) of fixed length with the best sum-of-pairs (SP) score (Zaslavsky and Singh, 2006). That is, given K sequences and a block length parameter ℓ, the goal is to find an ℓ-long subsequence (ℓ-mer) from each input sequence so that the total similarity among selected blocks is maximized, where similarity between the subsequences is defined by summing shared background-corrected identity along the sequence.

More formally, denote by $\mathcal{S}_i$ the set of all ℓ-mers in input sequence i. Given a similarity score $\mathrm{sim}(s_i, s_j)$ between pairs of ℓ-mers s_i, s_j, the objective is to maximize the SP-score of a motif defined in terms of the pairwise similarities:

$$\text{SP-score}(S) = \sum_{i<j} \mathrm{sim}(s_i, s_j) \tag{1}$$

where $S = (s_1, \ldots, s_K), s_i \in \mathcal{S}_i$ denotes a selection of ℓ-mers for each input sequence, and $\mathrm{sim}(s_i, s_j)$ is calculated by assigning a score of $\log(1/f(b))$ for a base b match, where $f(b)$ is the non-zero frequency of base b in the background [see Osada *et al.* (2004) for details], and 0 for any mismatch. Such a similarity computation combined with the SP-scoring scheme was shown to perform well in the context of motif finding (Hon and Jain, 2006; Zaslavsky and Singh, 2006).

2.2 MRF model

Since we have a discrete optimization problem and an objective function that is a sum of pairwise terms, we can transform the problem into a graphical model with pairwise potentials. Each model variable (or node in the graphical representation) corresponds to an input sequence, and the state of each node represents the selection of a particular position and corresponding ℓ-mer within the sequence (hereafter we use position and ℓ-mer interchangeably). To enable search for motif instances on the reverse-complemented strand, we extend the state space of each variable to include states corresponding to subsequences on that strand. We define the pairwise energies $E(s_i, s_j) = -\mathrm{sim}(s_i, s_j)$ for all pairs of node states. The existence of non-zero similarity scores between some states for all pairs of nodes results in a complete (fully connected) graphical model. With this definition, the probability of a given configuration, $P(S)$, is given by the following equation:

$$P(S) = \frac{1}{Z} e^{\frac{\text{SP-score}(S)}{T}} = \frac{1}{Z} e^{-\frac{\sum_{i<j} E(s_i, s_j)}{T}} \tag{2}$$

where Z is an explicit normalization factor and T is the system temperature (used as a free parameter, set to 1).

Finding the best motif with a single occurrence in each sequence under the SP-score is equivalent to identifying the most probable, or the maximum *a posteriori* (MAP), configuration of the model above. The MAP configuration can be obtained using a quantity known as max-marginals (MMs):

$$\mathrm{MM}_i(s_i) = \max_{S | S_i = s_i} P(S) \tag{3}$$

for which it can be shown (Pearl, 1988) that assignment of:

$$s_i^* = \arg \max_{s_i \in \mathcal{S}_i} \mathrm{MM}_i(s_i) \tag{4}$$

for each sequence yields the most probable motif selection s^*.

2.3 Belief propagation

The inference task of calculating the MMs and finding the MAP configuration in a graphical model is often addressed by the belief propagation (BP) algorithm and its variants (Pearl, 1988; Yedidia *et al.*, 2001). BP is a message passing algorithm that efficiently utilizes inherent locality in the graphical model representation. Messages are passed between neighboring (interacting) variables, and message contents describe one variable's 'belief' about its neighbor, based on their pairwise energy and the input of other messages. At a given iteration, assuming a complete graph and no singleton energies, the max product BP message, passed from variable i to variable j regarding j's state s_j is:

$$m_{i \to j}(s_j) = \max_{s_i} \left(e^{\frac{-E(s_i, s_j)}{T}} \prod_{k \neq j} m_{k \to i}(s_i) \right) \tag{5}$$

Messages are uniformly initialized and are iteratively recalculated ('passed') using Equation 5, until numeric convergence. Max-beliefs are then computed as the product of all incoming messages:

$$b_i(s_i) = \prod_k m_{k \to i}(s_i) \tag{6}$$

where $b_i(s_i)$ is the belief of state $s_i \in \mathcal{S}_i$, corresponding to some ℓ-mer in sequence i.

The BP algorithm was originally formulated for singly connected graphical models (i.e. when no 'loops' exist). For such graphs, the beliefs calculated in Equation 6 are equivalent to the MMs defined in Equation 3. For non-tree graphs the BP algorithm is not guaranteed to converge, and in theory even when it does converge, the obtained beliefs might differ significantly from the MMs. In practice, BP has been shown to be empirically successful in converging to optimal solutions when run on graphs with many cycles (e.g. Yanover and Weiss, 2003); we find this to also be the case in the current application.

2.4 Multiple solutions

The basic MRF framework presented above allows for finding a highly scoring motif with one occurrence per input sequence. While such motifs show good correspondence with known regulatory elements (Zaslavsky and Singh, 2006), extending the methodology to identifying multiple motif occurrences is desirable, as many transcription factors exhibit multiple binding locations upstream of genes (Karp *et al.*, 2007).

To extend the model to multiple motifs, consider the following example. Suppose that the data consist of $(K-1)$ sequences with a single binding site $s_i \in \mathcal{S}_i$ for a particular transcription factor within each, and one sequence, wlog sequence 1, with two such sites, denoted s_1 and s'_1. Then, the motifs $S=(s_1 \in \mathcal{S}_1, \ldots, s_K \in \mathcal{S}_K)$ and $S'=(s'_1 \in \mathcal{S}_1, \ldots, s_K \in \mathcal{S}_K)$ in the constrained 1-per-sequence formulation would both correspond to highly probable configurations of the MRF model.

Following this observation, we consider multiple highly probable configurations of the model to derive multiple motif occurrences. To obtain the M top-scoring solutions we apply the BMMF algorithm (Yanover and Weiss, 2004). At iteration m the algorithm determines the m-th solution, S^m. Conceptually, the next highest scoring configuration must differ from all previous top configurations in at least one sequence. As such, the BMMF algorithm partitions the solution space so that S^m and the next highest scoring solution are included in two different sub-spaces. The computed MMs are used to both infer solution S^m and guide the space partitioning. Pseudocode of the BMMF algorithm is presented in Supplementary Algorithm 1. Herein we describe in detail the first two iterations of the algorithm. At iteration 1 the algorithm chooses the assignment that maximizes the local beliefs [see Equation 4] as S^1. It then calculates the highest relative MM probability (termed BMM for 'best' MM) over every sequence i and position s_i within it, while excluding solution S^1:

$$\mathrm{BMM}^1 = \max_{i, s_i:\ s_i \neq S^1_i} \mathrm{MM}_i(s_i)$$

Let $\mathtt{i}^1$ and $\mathtt{s}_{\mathtt{i}^1}$ denote the sequence and position associated with BMM^1, respectively. The definition of MMs [Equation 3] implies that in S^2, the second best motif, the position associated with sequence $\mathtt{i}^1$ is $\mathtt{s}_{\mathtt{i}^1}$ (that is, $S^2_{\mathtt{i}^1} = \mathtt{s}_{\mathtt{i}^1}$). Therefore, the algorithm partitions the solution space into two sub-spaces:

(1) The sub-space in which sequence $\mathtt{i}^1$ is constrained to have a binding site at the maximizing position $\mathtt{s}_{\mathtt{i}^1}$ (by imposing the constraint $S_{\mathtt{i}^1} = \mathtt{s}_{\mathtt{i}^1}$). BMMF determines the next highest scoring solution S^2 and its next BMM (BMM^2) by running BP on that sub-space.

(2) The sub-space in which sequence $\mathtt{i}^1$ is constrained *not* to have a binding site at position $\mathtt{s}_{\mathtt{i}^1}$ ($S_{\mathtt{i}^1} \neq \mathtt{s}_{\mathtt{i}^1}$). BMMF runs BP on this sub-space (given the additional negative constraint) to compute the MMs and the corresponding BMM, associated with the next highest scoring solution within the sub-space.

The maximal BMM (obtained from either of these sub-spaces) is used to define the next partition and, subsequently, the next solution, S^3. Runs of BP are as described in Equations (5) and (6).

2.5 Statistical significance

The framework above gives us the ability to generate multiple solutions that correspond to highly probable configurations of the MRF model. Every solution represents a particular selection of ℓ-mers, one from each input sequence, and has a corresponding SP-score. We now assess the statistical significance of MRF configurations by comparing their corresponding scores to scores of motifs expected to arise from random data with the same characteristics. More specifically, to evaluate the significance of a solution with a particular SP-score σ, we calculate its e-value, or the expected number of solutions of equal or better quality. We first compute $P_\ell(X)$, the probability distribution of solution scores for ℓ-mers in K sequences in the following two steps. (1) We calculate the exact probability distribution $P_1(X)$ for a single column of K random bases [see Zaslavsky and Singh (2006) for details]. (2) Assuming independence between columns, we calculate the probability distribution $P_\ell(X)$ for ℓ random columns by convolution of $P_1(X)$ as in Tatusov *et al.* (1994), where we inductively construct a distribution for i columns based on the distribution for $(i-1)$ columns, $P_{i-1}(X)$, and the single column distribution $P_1(X)$.

Finally, to infer the e-value of score σ, we compute the probability that an ℓ-long pattern has a score greater than or equal to σ by chance alone, equal to $\sum_{x>=\sigma} P_\ell(x)$, and multiply it by the total number of possible motifs of length ℓ in the data. If the sequences have lengths $L_1, \ldots, L_K$, then the expected number of MSA solutions with score at least σ by chance alone, or the e-value, is equal to

$$e\text{-value}(\sigma) = \prod_i (L_i - l + 1) \times \sum_{x>=\sigma} P_\ell(x) \tag{7}$$

2.6 Aggregation and clustering

We enumerate 10 000 successive solutions by applying the BMMF algorithm. We then use the significance computation, and setting the e-value cutoff to 1 (though the algorithm is robust to other e-value thresholds in a reasonable range), consider all solutions with a score that falls above the significance threshold.

To aggregate the solutions and obtain multiple motifs, we make the following observations. (1) A group of solutions that exhibits a high degree of overlap, i.e. agreement on the motif positions for a majority of sequences, potentially point to multiple motif occurrences for sequences without absolute agreement. (2) Two seemingly different solutions may correspond to positive/negative shifts of the same motif. (3) Completely different non-overlapping groups of solutions may, indeed, indicate the presence of two entirely distinct motifs in the data.

As an example, consider the following few top solutions:

$$S^1 = (19, 122, 566, 6, 172, 93, 106, 87, 165, 250)$$
$$S^2 = (17, 120, 564, 4, 170, 91, 104, 85, 163, 248)$$
$$S^3 = (21, 124, 501, 8, 174, 95, 108, 89, 141, 252)$$

The notation lists consecutive solutions, indicating the positions of their corresponding motif occurrences in the respective input sequences. The first solution, S^1, finds the best motif instances to occur in position 19 of sequence 1, position 122 of sequence 2, etc. Applying the rules above, notice that solutions S^1 and S^2 are exact shifts by 2 of one another, and solution S^3 is part of the same cluster as well, since it agrees with solution S^1 on most sequences. Note, however, that S^3 differs from the others in binding site locations in sequences 3 and 9; this might indicate the presence of multiple motif occurrences in these sequences if support for those alternate sites is strong enough based on the solution ensemble in the cluster.

We consider shifts of up to half the length of the motif, ℓ, where a lower scoring solution is shifted to the higher scoring one (like S^2 shifting by two bases to be in phase with S^1). Solutions that agree (up to shift) on at least half of the sequences are clustered together; we discard clusters with less than K (number of input sequences) solutions.

Aggregating this information by sequence for each computed cluster, we get a histogram, specifying the number of times each binding site position within the sequence was observed among the solutions belonging to the cluster. Furthermore, since more highly ranked solutions are comprised of better scoring motifs, their histogram contribution is up-weighted accordingly. Specifically, we use a Boltzmann scheme and assign the weight of a putative motif position in solution S^m to:

$$w(S^m) = \alpha \cdot exp\left(\frac{\text{SP-score}(S^m)}{T}\right) \tag{8}$$

where T is an arbitrary temperature set to be 1% of SP-score(S^1) and α normalizes the sum of weights $\{w\}$ to M, which is the number of significant solutions considered, capped at 10 000. While ideally all solutions above the statistical significance threshold should be considered, every additional

solution is found at a cost in runtime. Since the weighting scheme assigns much lower weights to low-scoring solutions, considering at most 10 000 solutions provides a reasonable trade off between motif accuracy and algorithm runtime.

Multiple motif instances correspond to peaks in the histograms, and are readily evident among a vast background landscape (see Supplementary Fig. 1). To automatically detect the peaks, we use the criteria of minimal peak height, total weight and support (the weight of a peak relative to all remaining ones), as well as distance between peaks; various parameters settings obtain similar results.

2.7 Missing motifs

As a final step to our algorithm, we attempt to identify if any of the input sequences do not contain a motif occurrence. Our overall strategy is to find a group of sequences that might not have a motif instance, and compare the set of solutions in the original dataset to the one found after removal of the candidate sequences. Interestingly, the number of solutions below an equivalent *e*-value threshold in various subsets of the input data is a good indicator of the quality of the motif, and we use it as the criterion for deciding which subset of sequences to retain.

Consider the set of sequence positions contributing to an ensemble of solutions for a sequence with no motif occurrences. Since such a sequence has no 'consensus' position matching the motif, intuitively, the ensemble of positions is expected to be more diverse than that of sequences with motif instances. We therefore associate an entropy score with each such position ensemble and group these scores into three clusters (using complete linkage clustering): low, intermediate and high entropy clusters. We exclude the sequences placed in the high entropy cluster, and rerun BMMF on the remaining subset. If the number of significant solutions obtained by this new run of BMMF increases, we designate the left-out sequences as those missing motif instances and repeat this procedure, starting from the remaining subset of sequences; otherwise, we use the solutions for all current sequences to predict the motif. When less than 100 significant solutions are attained by BMMF (and entropy-based clustering might not be accurate), we compute the score contribution of each sequence to the top solution, remove the sequence with the lowest score, and rerun BMMF as before.

2.8 Dataset and performance metrics

We utilize a dataset consisting of collections of upstream regions for 36 *E.coli* transcription factors (Table 1, column 1) that is constructed from sets of experimentally derived binding sites cataloged by McGuire *et al.* (2000); Robison *et al.* (1998) and is described in detail in Osada *et al.* (2004).

To evaluate the quality of motif predictions, we employ some of the statistics used in a large-scale study by Tompa *et al.* (2005). These statistics (defined in Supplementary Materials), measure the degree of overlap between the predictions made by our approach and the known motifs at the nucleotide and site levels. The first measure, nucleotide performance coefficient (*nPC*), is a stringent statistic, penalizing a method for both failing to identify any nucleotide belonging to the motif (false negative) as well as falsely predicting any nucleotide outside the motif (false positive). The other statistic we consider is site-level sensitivity (*sSn*).

Table 1. Listing of *E.coli* transcription factor datasets (columns 1–4) and the details of motif finding by SAMF (columns 5–7)

TF	K	ℓ	Known TFBSs	Significant solutions	Motifs	Predicted TFBSs
ada[a]	3	31	3	2	1	3
araC	4	48	6	43	1	5
arcA	11	15	13	10 000	1	13
argR	8	18	17	10 000	1	15
crp	33	22	49	8912	1	60
cpxR	7	15	9	407	1	10
cytR	5	18	5	6	1	5
dnaA	6	15	8	526	1	9
fadR	5	17	7	23	2	5, 5
fis[b]	7	35	18	0	0	0
flhCD	3	31	3	106	3	3, 3, 3
fnr	10	22	12	10 000	1	16
fruR	10	16	11	863	1	16
fur	7	18	9	10 000	1	9
galR	6	16	7	10 000	1	10
glpR[a]	4	20	11	4	1	4
hns[b]	5	11	5	0	0	0
ihf[a]	19	48	24	1	1	19
lexA	17	20	19	10 000	1	22
lrp	4	25	14	7	1	4
malT	6	10	10	22	2	8, 8
metJ	5	16	15	2723	1	5
metR	6	15	8	46	1	7
modE	3	24	3	5	1	3
nagC	5	23	6	51	1	6
narL[a]	10	16	10	8	1	10
narP	7	16	7	76	2	7, 7
ntrC	4	17	5	492	1	7
ompR[b]	3	20	9	0	0	0
oxyR[b]	4	39	4	0	0	0
phoB	8	22	14	10 000	1	15
purR	18	26	20	10 000	1	25
soxS[a]	9	35	13	2	1	9
trpR	4	24	4	35	1	4
tus	5	23	5	10 000	1	5
tyrR	9	22	17	10 000	1	11

K is the number of sequences (each up to 600 bp long) that contain binding sites for the given TF. ℓ is the length of the motif [as reported by Robison *et al.* (1998)] searched for. *Known TFBSs* is the total number of known binding sites in dataset. *Significant solutions* is the number of MRF configurations passing the significance threshold (10 000 at most). *Motifs* is the number of distinct motifs produced by SAMF as the result of the clustering procedure. *Predicted TFBSs* is the comma-separated list of predicted numbers of binding sites in each cluster.

[a]Entries for which no clustering was performed by SAMF due to too few significant solutions; the top scoring solution is used as the motif prediction in this case.

[b]Entries for which no solution passed SAMF's significance threshold.

3 EXPERIMENTAL RESULTS

3.1 Performance evaluation

We apply SAMF to our dataset of 36 *E.coli* transcription factors (Table 1). For each transcription factor in the dataset, SAMF analyzes all statistically significant individual solutions, each of which requires two runs of BP (runtime up to 10 s, and much faster for majority of the datasets), and assigns them to distinct motifs by performing the clustering procedure detailed above. The solutions within clusters are then aggregated to determine all occurrences of the motif represented by the cluster. Thus, for every transcription factor SAMF predicts potentially more than one distinct motif. The number of distinct motifs found for each transcription factor is typically one or two, and the motif best overlapping with the known one is always within the top two clusters. This is consistent with the findings of Tompa *et al.* (2005) which suggest allowing each motif finder to predict multiple motifs for increased sensitivity.

We compare SAMF's performance on our dataset with the performance of three other methods, Weeder (Pavesi *et al.*, 2004), MEME (Bailey and Elkan, 1995) and MotifSampler

(Thijs *et al.*, 2001). We select these methods since they have been found to be the best performing among their algorithm type in the study of Tompa *et al.* (2005) applied to finding eukaryotic transcription factor binding sites. Together, they represent a broad range of motif finding techniques: Weeder is an enumerative algorithm, and MEME and MotifSampler are based on probabilistic position-specific scoring matrices (PSSMs), with MEME searching the space via Expectation Maximization and MotifSampler searching the space via Gibbs sampling. While MEME and MotifSampler can be run on the motif lengths found in our *E.coli* dataset, Weeder can only enumerate binding sites of length at most 12. Nevertheless, since Weeder is the best performing method in Tompa *et al.* (2005), we attempt to utilize it for uncovering bacterial transcription factor binding sites as well. Finally, in the performance evaluation, for each method we use the predicted motif that best corresponds to the known motif in the data.

Comparison to Weeder: Despite trying a range of parameters, we have found that Weeder is not effective in finding motifs on our prokaryotic dataset, as the algorithm predicts motifs vastly different from the known ones for all the test cases. The reason for the poor performance likely stems from the fact that, like most enumerative combinatorial methods, Weeder is not able to handle the lengths of the motifs in the dataset (which vary from 10–48). As a result, the motifs Weeder finds are vastly different from the known binding motifs (see Supplementary Fig. 2 for an example).

Comparison to MEME: We run MEME (Bailey and Elkan, 1995) with parameters to search for two distinct motifs, allowing any number of motif occurrences per sequence, and keeping all other parameters at their defaults. We disregard MEME's significance assessment of the found motifs, since otherwise no significant motifs are discovered for more than half the dataset, rendering the evaluation meaningless. To be fair, we allow SAMF to predict a motif in the four cases where no solutions pass its significance threshold (Table 1), using the top-scoring solution as the prediction.

The results of the performance comparison are shown in Fig. 1. Each bar in the chart measures the difference in *nPC* (Fig. 1A) or *sSn* (Fig. 1B) between SAMF and MEME. Considering both statistics, SAMF outperforms MEME, with the difference in performance between the two methods being statistically significant as measured by the Wilcoxon matched-pairs signed ranks test (*P*-values less than 0.0018 and 0.0043 for *nPC* and *sSn*, respectively). There are large differences notable for a few transcription factors. In particular, SAMF is able to find the *narL* and *narP* motifs almost completely, whereas MEME entirely misidentifies them. Another interesting case is that of *flhcd*, where SAMF's ability to find all solution clusters in the data aids motif discovery. Indeed, it is the second, less significant cluster that corresponds to the known motif.

Comparison to MotifSampler: We run MotifSampler (Thijs *et al.*, 2001) on our dataset using a provided *E.coli* background model. We set the parameters to look for two distinct motifs, allowing any number of motif occurrences per sequence. With these settings, MotifSampler returns motifs for only 17 of the 36 datasets, mostly the ones with a greater number of sequences. In contrast, SAMF finds significant motifs in 32 of the datasets. Even considering just the 17 transcription factors where both SAMF and MotifSampler find motifs, SAMF's average *nPC* is 0.48 and average *sSn* is 0.71; these numbers are 0.33 and 0.42 for MotifSampler. Overall, SAMF clearly outperforms MotifSampler on our dataset.

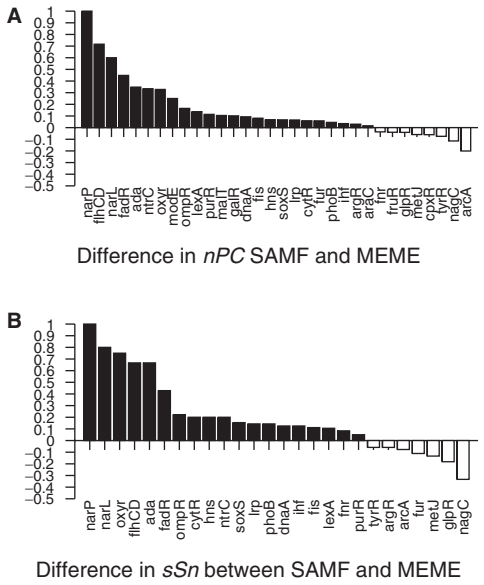

Fig. 1. Performance comparison for SAMF and MEME, given in terms of nucleotide performance coefficient (**A**) and site-level sensitivity (**B**). Significance assessment for both methods is disregarded. For every transcription factor dataset, the height of the bar indicates the difference in the metric, with bars above zero specifying better performance for SAMF and bars below zero for MEME. Plotted are only those datasets for which there is a difference in performance between the methods.

3.2 Sensitivity to noise

To evaluate SAMF's sensitivity to presence of noise, we perform motif finding on datasets, in which a number of the original sequences containing motif occurrences are randomly reshuffled. In particular, we have selected four transcription factor datasets of varying size and containing motifs of varying quality (as measured by information content). Applying motif finding to these noisy datasets, we track SAMF's performance using the *sSn* and *nPC* statistics as the fraction of random sequences varies between 0 and 0.5. SAMF's performance as measured by *nPC* (Fig. 2) remains steady with an increasing fraction of random sequences, and its overall ability to identify motifs in the data does not degrade considerably. The *sSn* trends are similar (data not shown). In one case SAMF fails to find the relatively poorly conserved *tyrR* motif as the noise rate approaches 0.5 and few of the sequences contain a motif instance.

We have also evaluated SAMF's sensitivity to correct motif length specification on the same sample datasets. When varying the motif lengths by up to 5 nt of the known binding site lengths (20–26 bp), we find practically no difference in the algorithm's ability to identify existing motifs (see Fig. 3 in Supplementary Materials).

3.3 Analysis of predictions

The performance evaluation of any method is only as good as the state of knowledge of the underlying biology. Indeed, some of SAMF's predictions, characterized as false positives, may be true binding sites, and can guide biological experiments. For instance, in the case of *phoB*, which is a signal transduction response regulator activated in phosphate depletion conditions, we correctly identify

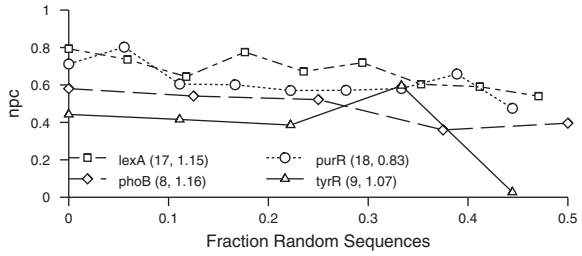

Fig. 2. SAMF's performance on datasets with noisy input sequences, given in terms of the nucleotide performance coefficient (*nPC*) plotted against fraction of random sequences. The *nPC* coefficients for motifs found by SAMF in the original data correspond to $x=0$. For each subsequent data point, the *nPC* statistic is computed based on the portion of the motif remaining in the data. Number of sequences in each dataset and motif information content are indicated following TF names in the legend.

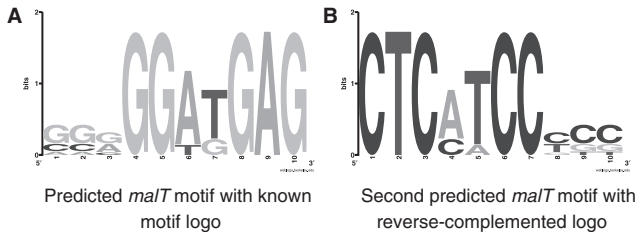

Fig. 3. Motifs predicted for the *malT* dataset. Shown in (**A**) is the prediction corresponding to the known *malT* motif logo, and in (**B**) the second predicted motif with the logo being the reverse complement of the known one.

almost all of its known targets in the dataset, and predict three additional ones. Two of these predictions, located upstream of genes *phoA/psiF* and *phoH*, are 11 bp away from a known binding site. Interestingly, structural studies of *phoB* effector domain complexed with its target DNA sequence reveal a tandem arrangement in which several *phoB* monomers bind head to tail to successive 11 bp sequences (Blanco *et al.*, 2002). A similar phenomenon is observed with another transcription factor, *arcA*, which is also a part of the larger *phoB/ompR* subfamily of response regulator proteins, and is known to regulate transcription by binding in tandem to target DNA sequences (Toro-Roman *et al.*, 2005). Three *arcA* sites that SAMF predicts and are deemed false positives, are, indeed, found 11 bp away from a known site.

We also analyze the additional motifs predicted by SAMF in alternate solution clusters. Using the STAMP tool (Mahony and Benos, 2007) and Pearson correlation coefficient as the distance measure, we compare alternate motifs against the *E.coli* database (Robison *et al.*, 1998). The most significant hit (*e*-value $4.36e^{-7}$) is produced for a motif found in the *fadR* dataset and matching the known *arcA* signature. This would imply that the transcription factor *arcA* is involved in regulating some of the known *fadR* targets. Indeed, when checked against the EcoCyc database (Karp *et al.*, 2007), an *arcA* binding site exists upstream of *fadR* regulated genes *fadB/fadA* and *fadD* that are in our *fadR* dataset, and a direct interaction between *arcA* and the promoters of genes in the *fadR* regulon was detected via ChIP analysis (Cho *et al.*, 2006).

Finally, we note an interesting finding for the *malT* dataset. *MalT* is a transcriptional activator that controls the expression of all the operons comprising the maltose regulon in *E.coli* (Larquet *et al.*, 2004). SAMF identifies 8 of 10 known binding sites and recovers the known *malT* motif logo in one of the two significant clusters (Fig. 3A). The other predicted motif follows a distinct pattern, with its occurrences separated by 22/23 bp either upstream or downstream from a known motif occurrence on the same strand. Moreover, the sequence logo of this motif is exactly the reverse complement of the known *malT* motif (Fig. 3B). Additionally, weaker sites are found alongside the known ones, providing evidence for sites in both direct repeat and inverted repeat configurations. The binding mechanism of *malT* to its diverse array of regulatory elements is still a subject of ongoing research (Larquet *et al.*, 2004); our discoveries may provide insights into that mechanism as well as the oligomeric structure for this transcription factor.

4 DISCUSSION

We have introduced a new formulation of motif finding based on MRFs. This framework allows us to use a BP-based algorithm to enumerate a set of distinct highly probable solutions. As a result, our algorithm SAMF is able to identify multiple binding sites in each input sequence, and can predict entirely distinct motifs as well. This approach is similar in spirit to a recent successful method (Reddy *et al.*, 2007), which produces a motif prediction based on multiple runs of a standard algorithm such as a Gibbs sampler. Compared with that approach, we explicitly constrain SAMF to find distinct solutions, which allows a faster exploration of the search space. Indeed, whereas the approach of Reddy *et al.* (2007) requires a few hours on a high-performance parallel computer to uncover yeast binding sites, our approach typically runs within a similar time range but on a standard desktop.

Since SAMF's performance depends upon the quality of the solutions it uncovers, we have investigated how well SAMF finds the optimal and successively near-optimal solutions to the underlying optimization problem. Accordingly, we compare the best solution found by BMMF against the globally optimal solution, earlier determined for most of the transcription factors in our dataset by a combinatorial optimization method (Zaslavsky and Singh, 2006). Indeed, for all but a single case, BMMF retrieves this optimal solution or finds a comparable one (e.g. a shift of the MAP solution). This is consistent with previous studies (Fromer and Yanover, 2009; Yanover and Weiss, 2004) showing that BMMF works well in practice and obtains a better set of top configurations than other algorithms, such as those based on Gibbs sampling. Though finding the optimal solution to the underlying optimization problem is an important goal, for any formulation, the quantity being optimized is a mathematical approximation of what we expect from biological motifs. We presume that obtaining multiple distinct solutions provides our algorithm with an additional benefit in that it enables SAMF to mitigate such weaknesses inherent in any formulation of motif finding.

We also note that our MRF framework is independent of the particular choice of sequence similarity function, and multiple such scoring schemes, as long as they are pairwise, can be applied. In particular, if it is known that a motif of interest is palindromic, as

is the case for many bacterial transcription factor binding sites, this can be incorporated at the level of the similarity function.

Overall, SAMF produces excellent results when applied to a dataset of *E.coli* transcription factors and their target upstream regions, finding most known binding sites better than other methods, and predicting novel sites whose veracity is supported by several lines of evidence. Interestingly, as part of SAMF's evaluation, we have found that methods that work well for detecting eukaryotic transcription factor sites (Tompa *et al.*, 2005) may need to be modified to be applied successfully for uncovering bacterial transcription factor binding sites. In particular, the length constraint is a major limitation of enumerative methods, and the significance evaluation of the PSSM-based methods may be too conservative. Finally, we note that while SAMF has been tested on a dataset of prokaryotic upstream regions for binding site discovery, it can be readily applied and tested in other motif finding settings as well.

Funding: Fred Hutchinson Cancer Research Center, Seattle, WA (in part); NIH Center of Excellence at Princeton University (P50 GM071508, in part); National Science Foundation (DGE-9972930, in part); National Institutes of Health award (HHSN266200500021C, in part); National Science Foundation (IIS-061223 to M.S.); National Institutes of Health (GM076275 to M.S.).

Conflict of Interest: none declared.

REFERENCES

Bailey,T. and Elkan,C. (1995) Unsupervised learning of multiple motifs in biopolymers using expectation maximization. *Mach. Learn.*, **21**, 51–80.

Blanco,A.G. *et al*. (2002) Tandem DNA recognition by PhoB, a two-component signal transduction transcriptional activator. *Structure*, **10**, 701–713.

Cho,B.K. *et al*. (2006) Transcriptional regulation of the fad regulon genes of *Escherichia coli* by ArcA. *Microbiology*, **152**, 2207–2219.

Das,M. and Dai,H. (2007) A survey of DNA motif finding algorithms. *BMC Bioinformatics*, **8**, S21.

Elemento,O. *et al*. (2007) A universal framework for regulatory element discovery across all genomes and data-types. *Mol. Cell*, **28**, 337–350.

Fromer,M. and Yanover,C. (2009) Accurate prediction for atomic-level protein design and its application in diversifying the near-optimal sequence space. *Proteins Struct. Funct. Bioinform.* (in press).

Hermsen,R. *et al*. (2006) Transcriptional regulation by competing transcription factor modules. *PLoS Comput. Biol.*, **2**, e164.

Hon,L.S. and Jain,A.N. (2006) A deterministic motif finding algorithm with application to the human genome. *Bioinformatics*, **22**, 1047–1054.

Hu,J. *et al*. (2005) Limitations and potentials of current motif discovery algorithms. *Nucleic Acids Res.*, **33**, 4899–4913.

Hu,J. *et al*. (2006) EMD: an ensemble algorithm for discovering regulatory motifs in DNA sequences. *BMC Bioinformatics*, **7**, 342.

Karp,P.D. *et al*. (2007) Multidimensional annotation of the *Escherichia coli* K-12 genome. *Nucleic Acids Res.*, **35**, 7577–7590.

Larquet,E. *et al*. (2004) Oligomeric assemblies of the *Escherichia coli* MalT transcriptional activator revealed by cryo-electron microscopy and image processing. *J. Mol. Biol.*, **343**, 1159–1169.

Lee,T. *et al*. (2002) Transcriptional regulatory networks in *Saccharomyces cerevisiae*. *Science*, **298**, 799–804.

MacIsaac,K. and Fraenkel,E. (2006) Practical strategies for discovering regulatory DNA sequence motifs. *PLoS Comput. Biol.*, **2**, e36.

Mahony,S. and Benos,P.V. (2007) STAMP: a web tool for exploring DNA-binding motif similarities. *Nucleic Acids Res.*, **35**, W253–W258.

McGuire,A. *et al*. (2000) Conservation of DNA regulatory motifs and discovery of new motifs in microbial genomes. *Genome Res.*, **10**, 744–757.

Mukherjee,S. *et al*. (2004) Rapid analysis of the DNA-binding specificities of transcription factors with DNA microarrays. *Nat. Genet.*, **36**, 1331–1339.

Osada,R. *et al*. (2004) Comparative analysis of methods for representing and searching for transcription factor binding sites. *Bioinformatics*, **20**, 3516–3525.

Pavesi,G. *et al*. (2004) Weeder web: discovery of transcription factor binding sites in a set of sequences from co-regulated genes. *Nucleic Acids Res.*, **32**, W199–W203.

Pearl,J. (1988) *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann.

Reddy,T.E. *et al*. (2007) Binding site graphs: a new graph theoretical framework for prediction of transcription factor binding sites. *PLoS Comput. Biol.*, **3**, e90.

Robertson,G. *et al*. (2007) Genome-wide profiles of STAT1 DNA association using chromatin immunoprecipitation and massively parallel sequencing. *Nat. Methods*, **4**, 651–657.

Robison,K. *et al*. (1998) A comprehensive library of DNA-binding site matrices for 55 proteins applied to the complete *Escherichia coli* K-12 genome. *J. Mol. Biol.*, **284**, 241–254.

Roth,F.P. *et al*. (1998) Finding DNA regulatory motifs within unaligned noncoding sequences clustered by whole-genome mRNA quantitation. *Nat. Biotechnol.*, **16**, 939–945.

Stark,A. *et al*. (2007) Discovery of functional elements in 12 Drosophila genomes using evolutionary signatures. *Nature*, **450**, 219–232.

Tatusov,R. *et al*. (1994) Detection of conserved segments in proteins: Iterative scanning of sequence databases with alignment blocks. *Proc. Natl Acad. Sci. USA*, **91**, 12091–12095.

Tavazoie,S. *et al*. (1999) Systematic determination of genetic network architecture. *Nat. Genet.*, **22**, 281–285.

Thijs,G. *et al*. (2001) A higher order background model improves the detection of regulatory elements by Gibbs Sampling. *Bioinformatics*, **17**, 1113–1122.

Tompa,M. *et al*. (2005) Assessing computational tools for the discovery of transcription factor binding sites. *Nat. Biotechnol.*, **23**, 137–144.

Toro-Roman,A. *et al*. (2005) Structural analysis and solution studies of the activated regulatory domain of the response regulator ArcA: a symmetric dimer mediated by the $\alpha4-\beta5-\alpha5$ face. *J. Mol. Biol.*, **349**, 11–26.

Webb-Robertson,B.J. *et al*. (2008) Measuring global credibility with application to local sequence alignment. *PLoS Comput. Biol.*, **4**, e1000077.

Wijaya,E. *et al*. (2008) MotifVoter: a novel ensemble method for fine-grained integration of generic motif finders. *Bioinformatics*, **24**, 2288–2295.

Yanover,C. and Weiss,Y. (2003) Approximate inference and protein-folding. In *NIPS 15*, MIT Press, Cambridge, MA, pp. 1457–1464.

Yanover,C. and Weiss,Y. (2004) Finding the M most probable configurations using loopy belief propagation. In *NIPS 16*. MIT Press, Cambridge, MA.

Yedidia,J.S. *et al*. (2001) Understanding belief propagation and its generalizations. In *IJCAI (distinguished lecture track)*.

Zaslavsky,E. and Singh,M. (2006) A combinatorial optimization approach for diverse motif finding applications. *Algorithms Mol. Biol.*, **1**, 13.