BIOINFORMATICS ORIGINAL PAPER Vol. 22 no. 24 2006, pages 2996–3002
doi:10.1093/bioinformatics/btl537

Sequence analysis

MUSA: a parameter free algorithm for the identification of biologically significant motifs

Nuno D. Mendes, Ana C. Casimiro, Pedro M. Santos[1], Isabel Sá-Correia[1], Arlindo L. Oliveira and Ana T. Freitas*

INESC-ID, Instituto Superior Técnico, Rua Alves Redol, 9 1000-029 Lisboa, Portugal and [1]Biological Sciences Research Group, Centro de Engenharia Biológica e Química, Instituto Superior Técnico, Av. Rovisco Pais, 1049-001, Lisboa, Portugal

Received on July 26, 2006; revised on September 29, 2006; accepted on October 13, 2006
Advance Access publication October 26, 2006
Associate Editor: Golan Yona

ABSTRACT

Motivation: The ability to identify complex motifs, i.e. non-contiguous nucleotide sequences, is a key feature of modern motif finders. Addressing this problem is extremely important, not only because these motifs can accurately model biological phenomena but because its extraction is highly dependent upon the appropriate selection of numerous search parameters. Currently available combinatorial algorithms have proved to be highly efficient in exhaustively enumerating motifs (including complex motifs), which fulfill certain extraction criteria. However, one major problem with these methods is the large number of parameters that need to be specified.

Results: We propose a new algorithm, MUSA (Motif finding using an UnSupervised Approach), that can be used either to autonomously find over-represented complex motifs or to estimate search parameters for modern motif finders. This method relies on a biclustering algorithm that operates on a matrix of co-occurrences of small motifs. The performance of this method is independent of the composite structure of the motifs being sought, making few assumptions about their characteristics. The MUSA algorithm was applied to two datasets involving the bacterium *Pseudomonas putida* KT2440. The first one was composed of 70 σ^{54}-dependent promoter sequences and the second dataset included 54 promoter sequences of up-regulated genes in response to phenol, as suggested by quantitative proteomics. The results obtained indicate that this approach is very effective at identifying complex motifs of biological significance.

Availability: The MUSA algorithm is available upon request from the authors, and will be made available via a Web based interface.

Contact: atf@inesc-id.pt

Supplementary information: An appendix is available at http://algos.inesc-id.pt/~atf under 'Papers on-line'.

1 INTRODUCTION AND RELATED WORK

Despite the remarkable success of computational biology tools in some areas of application like gene finding and sequence alignment, there are still problems for which no definitive methods have been developed. Notably, the accurate identification of biologically meaningful nucleotide sequences in *cis*-regulatory regions remains an open problem.

Motif finding is the problem of discovering promoter sequences and binding sites for transcription factors, usually referred to as consensus sequences or motifs, without any prior knowledge of their characteristics. These motifs can be sought by analyzing regulatory regions taken from genes of the same organism or from related genes of different organisms. Currently available methods can roughly be classified in two main classes: probabilistic and combinatorial. This classification covers most, although not all, popular motif finders currently available.

Probabilistic methods have the advantage of requiring few search parameters but rely on probabilistic models of the regulatory regions, which can be very sensitive with respect to small changes in the input data. Some of these methods also make simplifying assumptions about the nature and abundance of the motifs to be extracted.

The most popular probabilistic methods include approaches based on EM (Expectation-Maximization) (Segal and Sharan, 2005) like PROJECTION (Buhler and Tompa, 2002) and MEME (Bailey and Elkan, 1994) or its stochastic counterpart, Gibbs sampling (Schug and Overton, 1997) used by AlignACE (Roth *et al*., 1998), BioProspector (Liu *et al*., 2001) and GibbsDNA (Lawrence *et al*., 1993) These methods use a two-phase iterative procedure where in the first step the likeliest occurrences of the motif are identified, based on a model computed in the previous iteration. The second step adjusts the model for the motif (usually a weight matrix) based on the occurrences determined in the previous step. In the first iteration the parameters of the initial model are usually set randomly.

The major drawback with these algorithms is their sensitivity to noise in the data and the fact that they are not guaranteed to converge to a global maximum. Moreover, most of them assume that there will only be one motif occurring in the input sequences and at most once in each sequence. Some algorithms like MEME have removed these assumptions paying the price of being less efficient (Bailey and Elkan, 1994).

Combinatorial methods, which typically extract motifs consisting of plain nucleotide sequences or sequences over a degenerate alphabet, usually involve enumerating all possible patterns either explicitly or implicitly. The simplicity of this approach allows us to define a clear computational problem. Consider a set of sequences $\mathcal{S} = \{S_1, S_2, \ldots, S_t\}$. We want to find motifs within a range of

*To whom correspondence should be addressed.

lengths $l_{\min}, \ldots, l_{\max}$, which occur on $q \leq t$ of the presented sequences with at most e mismatches, i.e. having at most e nucleotide substitutions (also referred to as having a Hamming distance up to e). Combinatorial methods take one of two possible approaches. The first approach enumerates all possible patterns of a fixed length l (an l-mer) and verify its occurrence in the input sequences with at most e mismatches. The second approach takes each l-mer occurring in the input sequences and generates its e-mismatch neighborhood.

Several branch-and-bound algorithms have been proposed in the last few years that try to reduce the exponential search space taking advantage of sophisticated data structures. The MULTIPROFILER algorithm (Keich and Pevzner, 2002) follows a sample-driven approach whereby it manages to avoid generating the e-mismatch neighborhood for all sampled sequences. The WINNOWER algorithm (Pevzner and Sze, 2000) is based on a graph formulation. Motifs are found by identifying cliques in the graph. MITRA (Eskin and Pevzner, 2002) relies on a mismatch tree that partitions the search space. The algorithm will stop branching as soon as it determines that the subspace associated with a node is unable to hold a motif occurring in a least $q \leq t$ input sequences. SMILE (Marsan and Sagot, 2000) and RISO (Carvalho *et al*., 2006) use a generalized suffix-tree to represent the set of input sequences. They perform a lexicographic search to identify motifs which occur in $q \leq t$ input sequences with at most e mismatches.

Although combinatorial motif methods are able to find all motifs that match the specifications, they have the drawback that they are not good at discriminating the relevant extracted motifs from the potentially numerous false hits. They also require a large number of parameters to be specified. The problem of determining what portion of the output corresponds to a biologically significant result has been addressed mostly through the use of statistical techniques and biological reasoning and it is a challenge in its own right.

A key feature of modern motif finders is the ability to extract complex motifs, i.e. motifs with gaps or spacers, otherwise known as structured motifs or multi-ads (dyads, triads, etc.).

The advantages of considering complex motifs are many-fold. On the one hand, complex motifs can be better models of promoter regions. Some transcription factor DNA-binding domains have a composite structure, forming dimers (helix–loop–helix or leucine-zipper domains). The cooperative binding of several transcription factors and RNA-polymerase to the DNA molecule also seems to be bound by distance restrictions. On the other hand, many authors now agree that component motifs may be too weak to be extracted in isolation, i.e. they may be poorly distinguishable from the surrounding noise in the sequences. However, by imposing a certain distance between component motifs an unusual pattern may be identified.

In this paper we present a new method that can be used either autonomously, in the search for biologically relevant motifs, or, in alternative, used to adjust extraction parameters for modern motif finders. The method was validated both with synthetic and real data, and was used to discover new biologically significant motifs that have so far eluded searches performed using other motif finders. A biological analysis of the relevance of these new motifs is presented. See the Supplementary material for a comparison of the results obtained with competing approaches.

2 METHODS

Current methods for the extraction of complex motifs have a major drawback. Their output is, in practice, extremely sensitive with respect to the values of the input parameters. If we are too permissive by allowing a high degree of degeneration or by considering a large range of allowable lengths or yet, if we require the motifs to be present only in a small fraction of the input sequences, we may get an incommensurate number of motifs as output and we are left with the problem of identifying the biologically relevant ones. On the other hand, if we specify rigid parameters, like a specific length or low degree of degeneration and require the motif to be present in all sequences we may get no output at all. In fact, without any prior information, any rigid parameter specification is purely speculative. These problems are even more pressing for complex motifs where one needs to specify the number of components, the allowed length and the number of mismatches for each, and also the distance between components. An exhaustive search in the parameter space in this case is absolutely unfeasible.

Sagot pointed out (Marsan and Sagot, 2000) the need to distinguish between a motif and its occurrences in the input sequences. In fact, Sagot avoided the use of the term motif altogether introducing the notion of model. A model corresponds to a description of what constitutes a model occurrence. This distinction is particularly useful for complex motifs. A model (simple motif) is defined as a sequence over Σ^+, where Σ is the alphabet. A model m is said to have an e-occurrence, or simply an occurrence, in the input sequences, if there is a word u in the input sequences at a Hamming distance of m not greater than e. A model is said to be valid if it has occurrences in at least $q \leq t$ input sequences, where q is the quorum. A structured model (complex motif) is defined as a pair *(m, d)* where:

(1) $m = (m_i)_{1 \leq i \leq p}$ is a p-tuple of models ($m_i \in \Sigma^+$), denoting p components
(2) $d = (d_{\min_i}, d_{\max_i}, \delta_i)_{1 \leq i \leq p-1}$ is a $(p-1)$-tuple of triplets, denoting the $p-1$ gaps between components

Furthermore, considering a set of input sequences $\mathcal{S} = \{S_1, \ldots, S_t\}$, a structured model is said to be valid if, for all $1 \leq i \leq p-1$ and for all occurrences u_i of m_i, there are occurrences $u_1, \ldots, u_p$ of simple motifs $m_1, \ldots, m_p$ such that:

(1) $u_1, \ldots, u_p$ belong to the same input sequence
(2) there exists d_i, with $d_{\min_i} + \delta_i \leq d_i \leq d_{\max_i} - \delta_i$, such that the distance between the end position of u_i and the start position of u_{i+1} in the sequence is in $[d_i - \delta_i, d_i + \delta_i]$
(3) d_i is the same for the p-tuple of occurrences present in at least $q \leq t$ distinct input sequences

These definitions serve the purpose of a motif finder, which needs to restrict the search space and to decide what is a sufficiently common pattern so that it can be reported.

The purpose of the method proposed in this paper is to identify features in the input sequences that indicate the presence of interesting patterns by taking a broader view of the search space. The algorithm makes no assumptions about the number of components of the complex motifs it is looking for, nor about the distances between each component.

In this context, we define a complex motif as a number p of component simple motifs each separated by distances $(d_i, \varepsilon)_{1 \leq i \leq p-1}$, where p is the number of components. An occurrence of a complex motif is, similarly, a set of exact occurrences of each component simple motif, $u_1, \ldots, u_p$ in the same input sequence, where each occurrence is separated by a gap whose length is in $[d_i - \varepsilon, d_i + \varepsilon]$.

Both p and the distances d_i are determined by the algorithm. The only parameter we are expected to specify is ε.

S_1 : TAACCTGGTACA
S_2 : CGAATCTTGGTC
S_3 : GGAACTGCGGTG
S_4 : CTAATCCTAGGC
S_5 : GTAACTTCCGGT
S_6 : TCAAGCCTAGGC

Fig. 1. A set of input sequences

2.1 Matrix of co-occurrences

To avoid arbitrarily defined extraction parameters, we try to characterize certain features of the input sequences. To that effect, we begin by building a matrix of co-occurrences, $\mathcal{M}$.

To build this matrix we will first need to identify all occurrences of sequences of some small length, λ, i.e. all λ-mers in the input sequences, $\mathcal{S} = \{S_1, \ldots, S_t\}$.

Let $L(\mathcal{S}) = \{m_1, \ldots, m_z\}$ be the list of all such λ-mers, noting that $z \leq |\Sigma|^\lambda$. Figure 1 shows a set of input sequences that we will use to illustrate the definitions given below. In this example we will use $\lambda = 2$ to make it easier to follow.

DEFINITION 1. (List of occurrences of a λ-mer). *Let $\mathcal{S}$ be a set of input sequences and let $m \in L(\mathcal{S})$. $Occ_{i,\mathcal{S}}(m)$ denotes the set of coordinates of all occurrences of m in $S_i \in \mathcal{S}$.*

This set, $\mathrm{Occ}_{i,\mathcal{S}}(m)$, is therefore a list of integers denoting the positions at which we can find m on a sequence $S_i \in \mathcal{S}$. Whenever the set of input sequences, $\mathcal{S}$, is clear from the context we will simply write $\mathrm{Occ}_i(m)$. Concerning the example in Figure 1, it is easy to see that $\mathrm{Occ}_3(\mathrm{GG}) = \{1, 9\}$, $\mathrm{Occ}_1(\mathrm{AA}) = \{2\}$ or that $\mathrm{Occ}_6(\mathrm{TT}) = \varnothing$.

DEFINITION 2. (Configuration of a pair of λ-mers). *Let $\mathcal{S}$ be a set of input sequences. A configuration of a pair of λ-mers is a triple (m_r, m_s, d), with $m_r, m_s \in L(\mathcal{S})$ and $d \in Z \backslash \{0\}$.*

In this definition we simply introduce a mathematical object which we call a configuration. This object is associated with a set of input sequences and will be used to denote the co-occurrence of a pair of λ-mers in a specific relative position.

DEFINITION 3. (Configurations of a pair of λ-mers over a sequence $S_i \in \mathcal{S}$). *Let $\mathcal{S}$ be a set of input sequences and let m_r, $m_s \in L(\mathcal{S})$. $\Delta_{i,\mathcal{S}}(m_r, m_s)$ denotes the set of all configurations of m_r and m_s over $S_i \in \mathcal{S}$.*

$$\Delta_{i,\mathcal{S}}(m_r, m_s) = \{(m_r, m_s, d) : d = c_s - c_r, \; c_r \in \mathrm{Occ}_{i,\mathcal{S}}(m_r), c_s \in \mathrm{Occ}_{i,\mathcal{S}}(m_s), \; c_r \neq c_s\}$$

Note that if we consider the configuration (m_r, m_s, d) and if $d < \lambda$ then the configuration actually represents the occurrence of a $(\lambda + d)$-mer. It is also interesting to note that $\Delta_{i,\mathcal{S}}(m_s, m_r) = \{(m_s, m_r, d) : (m_r, m_s, -d) \in \Delta_{i,\mathcal{S}}(m_r, m_s)\}$. Once again, we will use $\Delta_i(m_r, m_s)$ every time the set of input sequences is clear from the context. Considering the previous example, we can observe that $\Delta_3(\mathrm{AA}, \mathrm{GG}) = \{(\mathrm{AA}, \mathrm{GG}, -2), (\mathrm{AA}, \mathrm{GG}, 6)\}$ or that $\Delta_6(\mathrm{AA}, \mathrm{TT}) = \varnothing$.

DEFINITION 4. (Score of a configuration of a pair of λ mer). *Let $\mathcal{S}$ be a set of input sequences and let m_r, $m_s \in L(\mathcal{S})$ and $d \in \mathbb{N}$.*

$\mu_{i,\mathcal{S}} : \Sigma^\lambda \times \Sigma^\lambda \times \mathbb{Z} \mapsto \{0, 1\}$ is the membership function of a configuration with respect to the set of all configurations of the two λ-mers on an input sequence $S_i \in \mathcal{S}$, defined as:

$$\mu_{i,\mathcal{S}}(m_r, m_s, d) = \begin{cases} 1 & \text{if} \quad (m_r, m_s, d) \in \Delta_{i,\mathcal{S}}(m_r, m_s) \\ 0 & \text{otherwise} \end{cases}$$

$\sigma_S : \Sigma^\lambda \times \Sigma^\lambda \times Z \mapsto \{0, \ldots, |\mathcal{S}|\}$ is the score function for a configuration and is defined as:

$$\sigma_{\mathcal{S}}(m_r, m_s, d) = \sum_{i=1}^{|\mathcal{S}|} \mu_{i,\mathcal{S}}(m_r, m_s, d)$$

The score of a configuration of a pair of λ mers is nothing more than the number of sequences where that particular configuration can be observed. Like in previous definitions, we will use $\sigma(m_r, m_s, d)$ without mentioning the set of input sequences whenever it is clear which set we are considering. In the example of Figure 1 we have $\sigma(\mathrm{AA}, \mathrm{GG}, 7) = 3$, since GG occurs seven positions after AA in sequences S_4, S_5 and S_6, yielding $\mu_{4,\mathcal{S}}(\mathrm{AA}, \mathrm{GG}, 7) = \mu_{5,\mathcal{S}}(\mathrm{AA}, \mathrm{GG}, 7) = \mu_{6,\mathcal{S}}(\mathrm{AA}, \mathrm{GG}, 7) = 1$.

DEFINITION 5. (ε-tolerant score of a configuration of a pair of λ-mers). *Let $\mathcal{S}$ be a set of input sequences and let m_r, $m_s \in L(\mathcal{S})$ and $\varepsilon \in \mathbb{N}_0$. The ε-tolerant score of a configuration $\sigma^\varepsilon_{\mathcal{S}} : \Sigma^\lambda \times \Sigma^\lambda \times \mathbb{Z} \mapsto \mathbb{N}_0$ is defined as:*

$$\sigma^\varepsilon_{\mathcal{S}}(m_r, m_s, d) = \sum_{i=1}^{|\mathcal{S}|} \max_{k=-\varepsilon,\ldots,\varepsilon} \mu_{i,\mathcal{S}}(m_r, m_s, d+k) \qquad (d \neq 0)$$

Furthermore, $\sigma^\varepsilon_{\mathcal{S}}(m_r, m_s, 0) = 0$.

The concept of ε-tolerant score of a configuration of a pair of λ-mers addresses the need to allow for a configuration to have slight variations. This removes the strictness of requiring a pair of λ-mers to co-occur at fixed relative positions in order to have a high score. This can be illustrated by the example shown in Figure 1 where $\sigma(\mathrm{AA}, \mathrm{GG}, 7) = 3$, $\sigma(\mathrm{AA}, \mathrm{GG}, 6) = 2$ and $\sigma(\mathrm{AA}, \mathrm{GG}, 5) = 1$ but $\sigma^1(\mathrm{AA}, \mathrm{GG}, 7) = 5$, $\sigma^1(\mathrm{AA}, \mathrm{GG}, 6) = 6$ and $\sigma^1(\mathrm{AA}, \mathrm{GG}, 5) = 3$. A 1-tolerant score is able to grasp the fact that the 2 mer AA co-occurs with GG in all input sequences at a distance of 6 ± 1 positions. Incidentally, $\sigma^1(\mathrm{AA}, \mathrm{GG}, 4) = 1$, despite the fact that $\sigma(\mathrm{AA}, \mathrm{GG}, 4) = 0$. This can be useful to describe patterns of co-occurrence that have a high ε-tolerant score even though they never actually occur in the input sequences.

DEFINITION 6. (Most common configuration of a pair of λ-mers). *Let $\mathcal{S}$ be a set of input sequences and let $m_r, m_s \in L(\mathcal{S})$. A configuration (m_r, m_s, d^*) is said to a most common configuration of the two λ-mers if, for every configuration (m_r, m_s, d), $\sigma_{\mathcal{S}}(m_r, m_s, d^*) \geq \sigma_{\mathcal{S}}(m_r, m_s, d)$.*

Furthermore, we say it is a ε-tolerant most common configuration if the same assertion holds for the ε-tolerant score.

The notion of most common configuration will be used to find the configuration or, indeed, the configurations with the highest score for a pair of λ mers. From the example shown in Figure 1 it is easy to see that the most common configuration for the pair (AA, GG) is $(\mathrm{AA}, \mathrm{GG}, 7)$. The 1-tolerant most common configuration is, however, $(\mathrm{AA}, \mathrm{GG}, 6)$.

We can now define a matrix of co-occurrences that gathers the information about the ε-tolerant score of the most common configuration of every pair of λ-mers.

DEFINITION 7. (Matrix of co-occurrences). *Let $\mathcal{S}$ be a set of input sequences. A matrix of co-occurrences over $\mathcal{S}$ with ε tolerance, $\mathcal{M}^\varepsilon_{\mathcal{S}}$, is the matrix where each of its elements aij is defined as:*

$$a_{ij} = \sigma^\varepsilon_{\mathcal{S}}(m_i, m_j, d^*),$$

where (m_i, m_j, d^*) is a ε-tolerant most common configuration of $m_i, m_j \in L(\mathcal{S})$ and $i, j = 1, \ldots, |L(\mathcal{S})|$.

The matrix of co-occurrences, $\mathcal{M}^1$, derived from the input sequences of Figure 1 is shown in Figure 2. We can see, by inspecting the matrix and the input sequences, that there are two configurations with the maximum 1-tolerant score: $(\mathrm{AA}, \mathsf{CT}, 3)$ and $(\mathrm{AA}, \mathrm{GG}, 6)$. In this case, it is easy to see that AA co-occurs with CT in all sequences at a relative distance of 3 ± 1 and with GG, also in all sequences at a relative distance of 6 ± 1.

The matrix of Figure 2 is symmetric. The next lemma, for which we omit the proof, shows that this is always the case.

	AA	AC	AG	AT	CA	CC	CG	CT	GA	GC	GG	GT	TA	TC	TG	TT
AA	0	3	2	2	1	3	2	**6**	2	2	**6**	4	3	3	3	2
AC	3	1	0	0	1	1	2	3	1	1	3	3	2	1	2	1
AG	2	0	1	1	1	2	0	2	0	2	2	0	2	2	0	0
AT	2	0	1	0	0	1	1	2	1	1	2	1	1	2	1	1
CA	1	1	1	0	0	1	0	1	0	1	1	1	1	1	1	0
CC	3	1	2	1	1	0	1	3	0	2	4	2	2	2	1	1
CG	2	2	0	1	0	1	0	2	1	1	2	2	1	1	1	1
CT	**6**	3	2	2	1	3	2	1	2	3	5	3	3	2	3	2
GA	2	1	0	1	0	0	1	2	0	1	2	2	0	1	2	1
GC	2	1	2	1	1	2	1	3	1	1	2	1	2	2	1	0
GG	**6**	3	2	2	1	4	2	5	2	2	1	4	3	3	3	2
GT	4	3	0	1	1	2	2	3	2	1	4	1	2	2	3	2
TA	3	2	2	1	1	2	1	3	0	2	3	2	2	2	1	1
TC	3	1	2	2	1	2	1	2	1	2	3	2	2	1	1	1
TG	3	2	0	1	1	1	1	3	2	1	3	3	1	1	1	1
TT	2	1	0	1	0	1	1	2	1	0	2	2	1	1	1	0

Fig. 2. Matrix of co-occurrences, M^1, for the input sequences of Figure 1.

Complex motif:	TTGCAn_5TATTA	
Configurations of 4-mers:	(TTGC,TGCA,1)	(TGCA,TTGC,-1)
	(TTGC,TATT,6)	(TATT,TTGC,-6)
	(TTGC,ATTA,7)	(ATTA,TTGC,-7)
	(TGCA,TATT,5)	(TATT,TGCA,-5)
	(TGCA,ATTA,6)	(ATTA,TGCA,-6)
	(TATT,ATTA,1)	(ATTA,TATT,-1)

Fig. 3. Configurations of 4 mers induced by the presence of a complex motif.

LEMMA 1. *A matrix of co-occurrences, $\mathcal{M}_{\mathcal{S}}^{\varepsilon}$ is symmetric, i.e. $a_{ij} = a_{ji}$ for every $i,j = 1,\ldots,|L(\mathcal{S})|$.*

The algorithm developed, in a first stage, computes the matrix of co-occurrences with ε-tolerance, for a fixed ε, with a time complexity of $O(N^2)$ where $N = \sum_{i=1}^{|\mathcal{S}|} |S_i|$. The total space requirements, for this stage, are in $O(N + |\Sigma|^{2\lambda})$. Further details on these complexity analysis are available in (Mendes, 2005).

2.2 Biclustering approach

As we have said, the matrix of co-occurrences gives us a view, for each pair of λ-mers, of the abundance of its most common configuration (or configurations). The next step is to try to combine these configurations to form larger patterns, possibly complex motifs. In doing so, we are guided by the score values computed during the construction of the matrix.

Consider the example in Figure 3. The presence of a complex motif with two components of length five each separated by a distance of 5 nucleotides induces 12 configurations of pairs of four different 4 mers. Let us suppose that this complex motif is present in exactly eight different input sequences. The score of each of these configurations is therefore no lower than 8. Admitting that none of these configurations occur in other input sequences, Figure 4 represents a sub-matrix of the matrix of co-occurrences that would be generated.

This example illustrates the basic principle of our approach to the inference of complex motifs using the matrix of co-occurrences. However, what we set out to do is the reverse of the reasoning shown in this example, i.e. we want to identify certain patterns in the matrix of co-occurrences that could indicate the presence of a complex motif.

We begin by characterizing the patterns we are looking for.

DEFINITION 8. (Diagonally-punctured bicluster in a matrix of co-occurrences). *A diagonally-punctured bicluster, B(I,Λ), in a matrix of co-occurrences $\mathcal{M}$ is a subset of the elements a_{ij} of $\mathcal{M}$ described by a pair (I,Λ), with $\Lambda \subseteq I$, and defined as:*

$$B(I,\Lambda) = \{a_{ij} : i \neq j, i \in I, j \in I\} \cup \{a_{ii} : i \in \Lambda\}$$

with $i,j = 1,\ldots,|L(\mathcal{S})|$.

A diagonally-punctured bicluster in a matrix of co-occurrences is, therefore, an object that roughly corresponds to a square sub-matrix of $\mathcal{M}$ except that the elements in the diagonal have an optional membership.

	ATTA	TATT	TGCA	TTGC
ATTA	0	8	8	8
TATT	8	0	8	8
TGCA	8	8	0	8
TTGC	8	8	8	0

Fig. 4. Sub-matrix induced by TTGCAn_5 TATTA, assuming that it occurs in eight input sequences.

This is an unconventional type of bicluster for two reasons. First, the columns that belong to the bicluster are entirely defined by the indexes of the rows (and vice versa) and second, the diagonal elements are not necessarily included in the set of elements of the bicluster. This is, arguably, not a bicluster at all but since we lack a more appropriate term we will still call it a bicluster bearing in mind its special characteristics.

DEFINITION 9. (*h*-valid diagonally-punctured bicluster in a matrix of co-occurrences). *An h-valid diagonally-punctured bicluster, $B(I,\Lambda)$, in a matrix of co-occurrences $\mathcal{M}$ is a diagonally-punctured bicluster such that $a_{ij} \geq h$ for every $a_{ij} \in B(I,\Lambda)$.*

The sub-matrix of Figure 4 illustrates this concept. We can think of it as a diagonally-punctured bicluster where *I* corresponds to the set of indexes of the 4-mers ATTA, TATT, TGCA and TTGC, and where $\Lambda = \emptyset$. If this is the case, we are in the presence of an eight-valid diagonally-punctured bicluster.

Let us recall that we are looking for patterns in the matrix of co-occurrences that can indicate the presence of a complex motif. We are interested in identifying diagonally-punctured biclusters that include as many elements of the matrix as possible and are *h*-valid for the highest value of *h* attainable. Such a bicluster would hopefully signal the presence of a complex motif in as many as *h* different input sequences. As we have remarked earlier, a simple motif is just a particular case of a complex motif and a diagonally-punctured bicluster could, in fact, indicate the presence of a simple motif of length greater than λ. For instance, the motif AAATT induces the following configurations of 4-mers[1]: $(\text{AAAT},\text{AATT},1)$ and $(\text{AATT},\text{AAAT},-1)$, which would correspond to a diagonally-punctured bicluster in the matrix of co-occurrences (provided the motif was frequent enough across the input sequences).

This approach thinks of complex motifs (and simple motifs) as compositions of configurations of λ-mers that will be shown in the matrix of co-occurrences in the form of diagonally-punctured biclusters. However, since we consider only the most common configurations of pairs of λ-mers some information can be lost. For example, an interesting motif could fall short from being identified if another motif composed by the same set of λ-mers (or a superset) is more frequent.

The second stage of the algorithm begins by considering the matrix of co-occurrences, $\mathcal{M}_{\mathcal{S}}^{\varepsilon}$, determined in the first stage, starting with the elements with the highest score, *h*. Since this matrix is symmetrical, our starting point is either an element on the main diagonal or a pair of elements from the upper and lower triangle, respectively. In either case, it is a diagonally-punctured bicluster. For each of these biclusters we will then greedily add rows/columns as long as the corresponding elements have a score not lower than the score of our initial elements. The same is performed for the diagonal elements. Elements which have already been included in a bicluster are not used as a starting point for ulterior biclusters. This way we are effectively seeking biclusters with high scores, which include as many matrix elements as possible. Note that this allows biclusters to include elements with a higher score than the score of the original elements. This can be translated in the fact

[1]It also induces 6 configurations of 3-mers, 12 configurations of 2-mers, etc. The impact of the choice of the value of λ will be discussed later.

that a configuration which occurs in h input sequences will *a fortiori* also occur in $h' < h$ input sequences. The algorithm will continue by iteratively considering elements with decreasing values of h until a specified minimum score is reached (see Supplementary material for details).

The algorithm will consider at most $|L(\mathcal{S})|^2 \leq |\Sigma|^{2\lambda}$ matrix elements as the starting point and to each of these initial biclusters will add at most $|L(\mathcal{S})| \leq |\Sigma|^{\lambda}$ rows/columns. At each tentative addition of rows/columns it will have to check whether the resulting bicluster is h-valid resulting in at most $|L(\mathcal{S})|^2 \leq |\Sigma|^{2\lambda}$ comparisons. This yields a time complexity of $O(|\Sigma|^{4\lambda})$. However, the larger the biclusters the less matrix elements will be used as a starting point, so this bound is not tight. Determining a tighter bound is quite difficult since the relation between the average size of the biclusters and the number of initial elements considered is not easily established due to the fact that different biclusters can effectively share many matrix elements.

This second stage of the algorithm is a heuristic approach to the problem in the sense that it may miss good solutions. Consider a matrix of co-occurrences and another matrix, with the same size, where each element holds the value 1 if the corresponding score in the matrix of co-occurrences is not below h and 0 otherwise. This binary matrix can be seen as a graph $G = (V,E)$ and searching for all largest diagonally-punctured biclusters in this new binary matrix is the same as searching for all maximal cliques in the corresponding graph. This problem is known to be NP-hard (Moon and Moser, 1965).

Our algorithm is, therefore, trying to solve the equivalent to the problem of enumerating all maximal cliques for each score h it considers.

Once the diagonally-punctured biclusters of interest are identified an automated procedure reconstructs the motif that induced the configurations that have been grouped.

2.3 Assembling families of motifs

MUSA includes a post-processing step that assembles a list of individual motifs into families of motifs, to simplify the analysis. This assembly is performed by looking for motifs that can be exactly superimposed to higher significance motifs by shifting one single base. In many cases, these other motifs are artifacts generated by the existence of larger motifs degenerated in some positions. This simple approach is very useful in reducing the number of motifs, leading to a more tractable output, easier to interpret by humans. The statistical significance of motifs is also computed using a variant of the method proposed in (Robin *et al*., 2002) (see Supplementary material for details).

The results obtained by this process are presented by reporting the most significant motif of the family (with a P-value smaller than 10^{-3}) and the quorum that results from the union of all the sequences that contain at least one motif in the family. This post-processing assembling of motifs is illustrated in Tables 1 and 2.

3 DISCUSSION

In this section we present and discuss the result of applying the proposed method to several datasets. The method was applied to both artificially generated (synthetic) datasets and to two real datasets. See the Supplementary material for the results obtained with the synthetic data.

For the real data, we choose a well characterized dataset, the σ^{54}-dependent promoter sequences of *Pseudomonas putida* KT2440 (http://www.promscan.uklinux.net/RpoN/promscan.outfile.66.html). For this dataset, a binding site, corresponding to a complex motif, has been determined before with high confidence (Barrios *et al*., 1999). The identification of DNA motifs in the promoter regions of coordinately expressed genes revealed by genome-wide expression methods, is currently an extremely important task. Having this in mind, a second dataset, including promoter sequences of *P.putida* KT2440 genes that are up-regulated in response to phenol, as suggested by quantitative proteomics (Santos *et al*., 2004), was also tested.

Table 1. Families of motifs reported by MUSA for the σ^{54}-dependent dataset

Number	Family	Quorum	P-value
1	TGGC(7)TTGC	40/70	8.8e–37
2	TTCGCGG(6)CCGC	11/70	3.1e–27
3	TGGCAC	70/70	3.3e–08
4	CGCGAAG	50/70	1.3e–06
5	TGGCATGG	22/70	1.0e–05
6	CCGCTCC	58/70	1.7e–05
7	TAACAAG	32/70	3.7e–05
8	GCTGGC	57/70	4.1e–05
9	AAGGTTT	50/70	0.1e–03
10	AAACCC	42/70	0.1e–03
11	CAAAACCC	53/70	0.2e–03
12	ACAAGAA	44/70	0.3e–03
13	GGTTTT	54/70	0.3e–03
14	TCAGTG	53/70	0.4e–03
15	TTTTAT	24/70	0.4e–03
16	GGCACAGC	21/70	0.7e–03
17	GAGCGGG	34/70	0.8e–03
18	GCCTGT	55/70	0.9e–03

Table 2. Families of motifs reported by MUSA for the phenol dataset

Number	Family	Quorum	P-value
1	GAACAGC	31/54	2.7e–06
2	TCGC(11)CTCC	6/54	2.9e–06
3	CCACT(63)CCGC	4/54	1.3e–05
4	TCAAGGC	26/54	5.6e–05
5	ACAGG	35/54	0.1e–03
6	GTAC(81)GGCC	4/54	0.1e–03
7	CAAGGC	32/54	0.2e–03
8	CGATGAC	34/54	0.2e–03
9	CCAGAATC	39/54	0.3e–03
10	CCTTGA	44/54	0.3e–03
11	TTCC(67)CGAC	4/54	0.3e–03
12	TACCTG	29/54	0.4e–03
13	TCTTCGC	32/54	0.5e–03
14	CCTGTG	40/54	0.6e–03
15	CCGCCAGG	41/54	0.6e–03
16	TCAGG	40/54	0.7e–03
17	CGCGGGCA	38/54	0.8e–03
18	AACTGGG	30/54	0.8e–03

3.1 Application to the σ^{54}-dependent promoter regions of *P.putida* KT22440

In this section we present results of the application of our method to a dataset, composed of 69 putative σ^{54}-dependent promoter sequences of *P.putida* KT2440 (http://www.promscan.uklinux.net/RpoN/promscan.outfile.66.html) and the well characterized σ^{54}-dependent promoter *Pu* (Lorenzo *et al*., 1995). Promoters

recognized by RNA-polymerase with the alternative σ^{54} factor are unique in having highly conserved positions around −24 and −12 nucleotides upstream from the transcriptional start site, instead of the typical, and less conserved, −35 and −10 boxes. Other authors (Barrios *et al*., 1999) reported a consensus sequence derived from a collection of 186 −24/−12 promoter elements from 47 bacterial species, including *P.putida* KT2440. This consensus sequence is: mrNrYTGGCACGNNNNTTGCWNNw where R stands for purines, Y for pyrimidines, W for A or T and N for any nucleotide. Therefore, the dataset used to test MUSA algorithm contains a very well annotated binding site that corresponds to a complex motif, as has been outlined by Cases and co-workers (Cases *et al*., 2003).

The results presented here were obtained using MUSA and considering the following values for the input parameters: $\lambda = 4$ and $\varepsilon = 1$ (default values). The selected motifs were ranked in accordance with their statistical significance. See the Supplementary information for a graphical representation of the co-occurrences matrix of motifs.

As described above, the method groups the motifs into families. The families of motifs that are statistically significant are listed in Table 1. In this table, family 1 corresponds to the core of the binding site for the σ^{54} factor. Families 3, 5 and 8 also correspond to variations of the first box of the σ^{54} factor. In fact, the ability to report motif variations in different families is one of the interesting features of the algorithm. For example, if we consider the first box of family 1 (Table 1), TGGC, as the motif core, the variations in this box reported in family 3, 5 and 8, reflect the most conserved nucleotides present in the neighborhood of the core.

σ^{54}-dependent promoters are characterized not only by the presence of a well conserved −24/−12 nt sequence but also for their dependence on EBPs (enhancer binding proteins) for proper functioning (Studholme and Dixon, 2003). In nearly all σ^{54} EBPs investigated so far, there is a DNA-binding domain containing a helix–turn–helix sequence motif, enabling the protein to bind to specific DNA enhancer elements upstream of σ^{54}-dependent promoters, also known as upstream activation sequences (UAS) (Morett and Buck, 1989). Furthermore, in many of the σ^{54}-dependent promoters that have been investigated, global transcription factors, such as the integration host factor (IHF), play a key role on their regulation. It is likely, therefore, that very common motifs may be associated with DNA-binding sites for global transcription factors. This may be the case of the families of motifs 4, 6, 9, 11, 13, 14 and 18, that represent very common motifs present in, at least, 50 sequences. On the other hand, due to the very diverse biological functions of the genes with σ^{54}-dependent promoters (e.g. PP4867, branched-chain amino acid ABC transporter; PP4391-flagellar basal-body rod protein FlgB) it is reasonable to hypothesize that different EBPs would require different cis elements. It is, therefore, possible that the families representing less common motifs may correspond to different binding sites (UAS) for distinct EBPs. Among the different motif families, family 2 has the lower quorum but the second highest *P*-value. This family corresponds to a palindromic motif that we believe is of biological significance, possibly as a target of an EBP or as target of a specific transcription factor for those genes. All the results reported by this algorithm can be analyzed directly, or, in alternative, can be used to select the appropriate parameters for a combinatorial motif finder. For example, the motif in family 1 suggests that a search for motifs with two size four boxes, separated by (roughly) 7 nt should be performed. If a search with those parameters and one allowed error in each box was performed using a commonly used motif finder, it would retrieve the σ^{54} motif, present in all sequences.

3.2 Application to a dataset of *P.putida* KT2440 up-regulated genes in response to phenol

We have also applied MUSA to the analysis of a set of *P.putida* KT2440 genes that are up-regulated following cell exposure to phenol, as suggested by results from quantitative proteomics (Santos *et al*., 2004). This dataset included 54 promoter regions that control the expression of genes involved in different biological functions. The results of MUSA, shown in Table 2, were compared with known consensus sequences for the binding of various bacterial transcription factors, even though most of the consensus sequences reported in the literature have not been validated in *P.putida*. Tables 1 and 2 were compared in order to search for common motifs reported by MUSA. Family 18 motif, GCCTGT from Table 1, overlaps family 14 motif, CCTGTG from Table 2 (with one base shift). These motif families are very common, being present in over 74% of the promoter regions included in the two datasets, suggesting that it can be a possible target for a global regulator involved in stress response. Consistent with this hypothesis is the fact that the first box of the consensus sequence for LexA, CTGT{8} ACAG (Li *et al*., 2004), is included in the reported motifs of family 18 (Table 1) and family 14 (Table 2). LexA protein is the common repressor of the SOS genes, whose products are involved in DNA repair or allow the cell to tolerate DNA lesions until DNA repair is accomplished (Callero *et al*., 1991). The second box of the LexA consensus is included in family 16 of Table 1 and in families 1 and 5 of Table 2. The LexA control of a number of σ^{54}-dependent promoters (first real dataset analyzed) or of phenol responsive promoters (second real dataset analyzed) was never suggested before. However, the type of promoters examined in our study may respond to nutrient starvation and/or chemical aggression. It is, therefore, feasible that the regulatory protein LexA may play a role in the modulation of specific adaptive mechanisms suitable to overcome the referred stresses. For the families of motifs only present in Table 2, one particular family has attracted our attention. Family 10 motif, CCTTGA, matches with the consensus sequence TNtCNCcCTTGAA{13,15}CCCCATtTA reported by Cowing *et al*. (1985) for the σ^{32} transcription factor (also known as RpoH). σ^{32} is known to stimulate not only the transcription initiation from heat shock promoters but also from solvent catabolic promoters, such as Pm (Marques *et al*., 1999). Therefore, the possibility that σ^{32} may control the expression of a subset of genes involved in the response to chemical stress caused by phenol is a reasonable hypothesis that, certainly, deserves experimental confirmation.

4 CONCLUSIONS AND FUTURE WORK

In this paper, we present an effective method to find biologically significant motifs that can be used standalone or as a tool to determine the parameters required to run motif finders already available. The basic algorithm is coupled with a statistical significance assessment method, and with a post-processing step that assembles motifs into families of motifs leading to a more tractable output. Future work will include a more accurate assembly of the motifs and the generation of a PSSM representation of the families.

Although the method can handle degenerate motifs, it works well only as long as sufficiently conserved regions remain present. An important improvement would be a modification in the algorithm to make it able to cope with highly degenerated motifs, i.e. with higher rates of nucleotide substitutions.

Experiments performed with MUSA using promoter sequences from a bacterial experimental model, with the genome sequence available and results obtained from genome-wide expression analysis, led to the discovery of interesting motifs that are biologically relevant and that have eluded previous searches. The information that emerged from this analysis using the new algorithm proposed is certainly of interest to guide experimental research in the field of gene expression regulation in bacteria.

ACKNOWLEDGEMENT

This work was partially supported by projects POSI/SRI/47778/2002, Biogrid, POSI/EIA/57398/2004, DBYeast and POCI/BIO/56838/00, managed by FCT and MCTES.

Conflicts of Interest: none declared.

REFERENCES

Mendes,N.D. (2005) Inference of complex motifs using biclustering techniques. Master's thesis, IST, Technical University of Lisbon.

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

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

Eskin,E. and Pevzner,P.A. (2002) Finding motifs n the twilight zone. In *proceedings of RECOMB*, ACM Press, pp. 195–204.

Keich,U. and Pevzner,P.A. (2002) Finding motifs in the twilight zone. In *Proceedings of RECOMB*.

Marsan,L. and Sagot,M-F. (2000) Algorithms for extracting structured motifs using a suffix tree with an application to promoter and regulatory site consensus identification. *J. Comput. Biol.*, **7**, 345–360.

Moon,J.W. and Moser,L. (1965) On cliques in graphs. *Israel J. Math.*, **3**, 23–28.

Morett,E. and Buck,M. (1989) *In vivo* studies on the interaction of RNA polymerase-sigma 54 with the *klebsiella pneumoniae* and *rhizobium meliloti* nifH promoters. the role of NifA in the formation of an open promoter complex. *J. Mol. Biol.*, **210**, 65–77.

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

Schug,J. and Overton,G.C. (1997) Modeling transcription factor binding sites with Gibbs sampling and minimum description length encoding. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, **5**, 268–271.

Segal,E. and Sharan,R. (2005) A discriminative model for identifying spatial cis-regulatory modules. *J. Comput. Biol.*, **12**, 822–834.

Studholme,D.J. and Dixon,R. (2003) Domain architectures of sigma54-dependent transcriptional activators. *J. Bacteriol.*, **185**, 1757–1767.

Barrios,H. *et al.* (1999) Compilation and analysis of σ-54-dependent promoter sequences. *Nucleic Acids Res.*, **27**, 4305–4313.

Callero,S. *et al.* (1991) One-step cloning system for isolation of bacterial lexa-like genes. *J. Bacteriol.*, **173**, 7345–7350.

Carvalho,A.M. *et al.* (2006) An efficient algorithm for the identification of structured motifs in DNA promoter sequences. *IEEE Trans. Comput. Biol. Bioinform.*, **3**, 126–140.

Cases,I. *et al.* (2003) The sigma54 regulon (sigmulon) in *Pseudomonas putida*. *Environ. Microbiol.*, **5**, 1281–1293.

Cowing,D.W. *et al.* (1985) Consensus sequence for *Escherichia coli* heat shock gene promoters. *Proc. Natl Acad. Sci. USA*, **82**, 2679–2683.

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

Li,H. *et al.* (2004) Identification of the binding sites of regulatory proteins in bacterial genomes. *Proc. Natl Acad. Sci. USA*, **99**, 11772–11777.

Liu,X. *et al.* (2001) Bioprospector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pac. Symp. Biocomput.*, **6**, 127–138.

Lorenzo,V. *et al.* (1991) An upstream XylR- and IHF-induced nucleoprotein complex regulates the sigma 54-dependent Pu promoter of TOL plasmid. *EMBO J.*, **10**, 1159–1167.

Marques,S. *et al.* (1999) The XylS-dependent Pm promoter is transcribed in vivo by RNA polymerase with sigma 32 or sigma 38 depending on the growth phase. *Mol. Microbiol.*, **31**, 1105–1113.

Robin,S. *et al.* (2002) Occurrence probability of structured motifs in random sequences. *J. Comput. Biol.*, **9**, 761–774.

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.

Santos,P.M. *et al.* (2004) Insights into *Pseudomonas putida* KT2440 response to phenol-induced stress by quantitative proteomics. *Proteomics*, **4**, 2640–2652.