In silico representation and discovery of transcription factor binding sites

Giulio Pavesi, Giancarlo Mauri and Graziano Pesole

Abstract

Understanding the complex mechanisms governing basic biological processes requires the characterisation of regulatory motifs modulating gene expression at transcriptional and post-transcriptional level. In particular, extent, chronology and cell-specificity of transcription are modulated by the interaction of transcription factors with their corresponding binding sites, mostly located near (or sometimes quite far away from) the transcription start site of the gene. The constantly growing amount of genomic data, complemented by other sources of information such as expression data derived from microarray experiments, has opened new opportunities to researchers in this field. Many different methods have been proposed for the identification of transcription factor binding sites in the regulatory regions of co-expressed genes: unfortunately this is a very challenging problem both from the computational and the biological viewpoint. This paper provides a survey of existing methods proposed for the problem, focusing both on the ideas underlying them and their availability to the scientific community.

Giulio Pavesi
is assistant professor of Computer Science at the University of Milan. His research interests are mainly focused on bioinformatics in general, and regulatory motif discovery in particular. He also works on discrete models of complex systems.

Giancarlo Mauri
is full professor of Computer Science at the University of Milan-Bicocca. His research interests are mainly in the area of theoretical computer science, and include: formal languages and automata, computational complexity, computational learning theory, neural networks, cellular automata and bioinformatics.

Graziano Pesole
is full professor of Molecular Biology at the University of Milan, leading the Laboratory of Bioinformatics and Comparative Genomics. His research interests include bioinformatics tools for genome annotation and molecular evolution.

Keywords: *motif discovery, promoter, position-specific weight matrix, genome annotation, software tools, transcription factor binding site, transcription regulation*

Graziano Pesole,
Department of Biomolecular Science and Biotechnology,
University of Milan,
via Celoria, 26,
20133 Milano, Italy

Tel: +39 02 50314915
Fax: +39 02 50314912
E-mail: graziano.pesole@unimi.it

INTRODUCTION

One of the greatest challenges facing modern molecular biology is the understanding of the complex mechanisms regulating gene expression. In particular, extent, chronology and cell-specificity of transcription are modulated by the interaction of transcription factors (TFs) with their corresponding binding sites (TFBS), mostly located nearby the transcription start site (TSS) of the gene (ie proximal promoter region) or further apart (enhancers, silencers, etc).[1,2] The constantly growing amount of genomic data (complemented by other sources of information such as full-length cDNA sequencing projects[3,4] that permit the precise mapping of the TSS on the genome sequence) as well as expression data derived from microarrays and other experiments, have opened new opportunities to researchers.

Hence, the need for efficient and reliable methods for detecting novel *motifs* (or *signals*), that are significantly over-represented in the regulatory regions of sets of genes sharing common properties (eg expression profile, biological function, product cellular localisation). These motifs could in turn correspond to binding sites for some common TF(s) regulating the genes. Unfortunately, binding sites of the same TF are generally short (usually less than 12–14 base pairs, bp, long) and degenerate (similar but not identical) oligonucleotides, and this fact makes their computational discovery and large-scale annotation significantly harder. The problem is further complicated by the size of the sequences to be examined, which ranges from a few hundreds of nucleotides for a 'simple' organism such as yeast, to human regulatory regions, where TFBSs can be located several kilobases away from the TSS and on both sides.

This paper provides, without the claim of being exhaustive, a survey of a number of different methods and approaches to the problem, focusing in particular on those algorithms whose implementation is available to the scientific community, either via a web interface or by free download of the program. In any case, we also provide references to articles where

Article number = K184

strategies and methods are described in a more detailed way. We will not discuss here other different flavours of the problem, for example how to determine whether a set of sequences contains already known binding sites for some TF, or phylogenetic footprinting, for which we refer the reader, respectively, to Rahmann *et al*.[5] and Bulyk,[6] and references therein.

THE PROBLEM

To get things started, the problem can be formulated as follows. There is a set of regulatory regions taken from a set of genes likely or somehow known to be regulated by the same transcription factor(s). Ideally, there would be a some crystal ball that, taking the sequences, outputs something like 'the binding sites recognised by the common TF(s) are these, and these are their locations in the regulatory regions'. Of course, things are not this simple, from the very beginning. For example, it is quite hard to be sure that *every* gene of the set is regulated by the same transcription factor. But, as we will see, the problem is complicated enough even when we have a 'perfect' input, that is, all the sequences considered actually interact with the same TF(s). A similar analysis can be performed on whole genomes: That is, all (or a large number of) the regulatory regions of an organism are considered, and over-represented oligos can be suspected to play some role in the regulation of the genes, and therefore considered to be candidate TFBSs.

Virtually all the methods proposed so far are based on a few fundamental steps:

- First, one or more groups of oligonucleotides, similar enough to one another to be recognised by the same TF, are detected in the input sequences; these are candidate *motifs*.

- Each group is evaluated from the statistical point of view, to have an estimate on how 'surprising' it is to find such a group in the input; this measure should consider both the size of the group and how conserved it is, that is, how many times the oligos are found in the sequences and how much they differ from one another.

- The most significant motifs are output, and the oligos forming each one are the best candidate TFBSs for the same TF.

Modelling motifs

For the first step, it is necessary a method to *describe* a set of similar oligonucleotides, representing binding sites for the same TF. Clearly, this choice will influence the rest of the process, that is, the strategies used to find and evaluate the best groups. Two main approaches have been introduced for this task: represent the set of oligos with a *consensus*, or describe them with their alignment, expressed with a *profile* (also found in literature under the names *frequency matrix, position-specific score matrix, position specific weight matrix*). A simple example is shown in Figure 1. A consensus describes a set of oligos with the most frequent nucleotide in each position; thus, we can denote a set of TFBSs with a single oligonucleotide, but we have to somehow specify how and how much other oligos can differ from the consensus in order to be considered valid binding sites for the same TF. Sometimes, however, some positions of binding sites do not show a definite preference for a nucleotide, but rather admit more than one base or even any base. Thus, instead of limiting the consensus description to a single nucleotide for each position and allowing a maximum number of substitutions, another strategy is to incorporate the different alternatives in the description itself, by using ambiguous symbols. For example, GAL4 binding sites in yeast can be summed up as CGGNNNNNNNNNNCCG,[7] where N stands for any nucleotide. In this way, substitutions are allowed only in the core of the motif instances, that is, all

>YAL038W ATTCC
>YAL038W CTTCC
>YAL038W CTTCC
>YCR012W CTTCC
>YCR012W CTTCC
>YCR012W CTTCC
>YDR050C CATCC
>YDR050C CATCC
>YDR050C CTTCC
>YDR050C CTTCC
>YHR174W CATCC
>YOL086C CTTCC

> GCR1 frequency matrix and consensus

	1	2	3	4	5
A	0.08	0.25	0.0	0.0	0.0
C	0.92	0.0	0.0	1.0	1.0
G	0.0	0.0	0.0	0.0	0.0
T	0.0	0.75	1.0	0.0	0.0
Consensus	C	T	T	C	C

Figure 1: A set of binding sites for yeast TF GCR1 (taken from SCPD[7]), represented with a frequency matrix and the corresponding consensus.

oligos starting with CGG and ending with CCG, with any 11 nucleotides in the middle, are considered GAL4 binding sites. Other symbols can be used for describing any combination of nucleotides, as shown in Table 1. Thus, YYRRNAA, for example, stands for 'any oligo made of pyrimidine–pyrimidine–purine–purine–any nucleotide–adenine-adenine'. This method is used to describe binding site consensuses in TF databases such as TRANSFAC[9] and the SCPD.[7] While more powerful than single-nucleotide consensuses, these descriptions should be however taken with a grain of salt. The set of binding sites of Figure 1 could be represented with CWTCC or MWTCC. Assuming the list of sites in Figure 1 to be comprehensive (ie including all possible binding sites recognised by the same TF), the valid binding site ATTCC would not be included in the first consensus representation, differently from AATCC, not a valid binding site, which would instead be included in the second one.

A more flexible modelling solution is offered by *profiles,* obtained by aligning the TFBS instances, and by describing their alignment with the frequency of each nucleotide in each column of the alignment itself.[10–12] The result is a $4 \times m$ matrix, where m is the length of the oligos aligned (without gaps) and where the sum of each column equals 1. In this way, ambiguous positions where TFBSs admit different alternative nucleotides are implicitly expressed in the respective column of the matrix, as well as 'a preference', expressed by nucleotide frequency. Also, the matrix can be used to evaluate if, and how much, any oligo can be considered an instance of the sites described by the matrix itself. On the other hand, a TFBS consensus can be seen as a 'majority vote' on each of the columns of the corresponding profile.

It has to be noted, however, that a comprehensive list of binding sites is generally not available, and that the major goal of motif representation is to predict sites that have not been observed previously. Of course, the more comprehensive is the list of valid binding sites, the more representative we can expect to be its profile (or consensus) description as well as its prediction effectiveness.

Table 1: IUPAC-IUB recommended codes[8] used to denote ambiguous positions in nucleotide sequences

IUPAC	Nucleotides	Mnemonics
A		Adenine
C		Cytosine
G		Guanine
T		Thymine
R	A or G	puRines
Y	C or T	pYrimidines
W	A or T	Weak hydrogen bonding
S	G or C	Strong hydrogen bonding
M	A or C	aMino group at common position
K	G or T	Keto group at common position
H	A,C,T	not G
B	C,G,T	not A
V	A,C,G	not T
D	A,G,T	not C
N	A,C,G,T	aNy

Modelling the background

All in all, the most challenging problem in the discovery of TFBSs is perhaps finding a good statistical measure able to reflect the biological relevance of the different motifs and to discriminate those describing real binding sites from those that are the effect of random similarities.

Table 2: Name and web address of motif-finding programs (suitable or explicitly designed for TFBSs), available free of charge on the internet as of February 2004

Name	Address	Exec	Rep.	Ref.	ML	Q	A	C
AlignACE	http://atlas.med.harvard.edu/	Yes	P	47	R	R		
ANN SPEC	http://www.cbs.dtu.dk/services/DNAarray/ann-spec.php		P	48	R			
Bioprospector	http://bioprospector.stanford.edu/	Yes	P	59	R			Yes
CONSENSUS	http://stormo.wustl.edu/Consensus_Server.	Yes	P	42	R			
Co-Bind	http://ural.wustl.edu/softwares.html	Only	P	68	R			Yes
Gibbs Sampler (recursive sampler)	http://bayesweb.wadsworth.org/gibbs/gibbs.html	Yes	P	63				
GLAM	http://zlab.bu.edu/glam/	Only	P	62				
MDscan	http://bioprospector.stanford.edu/MDscan/index.html	Yes	P	80	R	R		
MEME	http://meme.sdsc.edu/meme/website/intro.html	Yes	P	43				
MIRA	http://compbio.ornl.gov/mira/cgi-bin/newForm.cgi		P	26	R	R		
MITRA	http://fluff.cs.columbia.edu:8080/domain/mitra.html	Yes	P	56	R			Yes
MobyDick	http://genome.ucsf.edu/mobydick/		C	38				
Motif Sampler	http://www.esat.kuleuven.ac.be/~thijs/Work/MotifSampler.html	Yes	P	60	R			
Multiprofiler	http://www-cse.ucsd.edu/groups/bioinformatics/software.html	Only	P	57	R			
P-Branching	http://www-cse.ucsd.edu/groups/bioinformatics/software.html	Only	P/C	58	R		R	
REDUCE	http://bussemaker.bio.columbia.edu/reduce		C	79	R			
RSA Tools*	http://rsat.ulb.ac.be/rsat/		P/C	34	R			
SMILE	http://bioweb.pasteur.fr/seqanal/interfaces/smile.html	Yes	C	31	R	R	R	Yes
Verbumculus	http://www.cs.ucr.edu/~stelo/Verbumculus/	Yes	C	30				
Weeder Web	http://www.pesolelab.it/(Tools link)		C	32				
YMF	http://abstract.cs.washington.edu/~saurabh/YMFWebRSH/YMFInput.pl	Yes	C	36	R		R	

The table lists from left to right whether the executables of the program can be downloaded and installed on a local machine (in the other cases, the program authors have to be contacted), the reference to the article describing the program and its usage, the type of binding site representation used (P = profile matrix, C = consensus), and the parameters that are required (R) to start the program: ML (motif length), Q (quorum, number of sequences that have to contain a motif), and A (degree of approximation, ie number of mutations allowed). Finally, the 'C' column lists programs that can find 'composite' motifs explicitly.

* RSA Tools is a series of pipelined tools for regulatory sequence analysis that includes facilities for regulatory sequence download, oligonucleotide frequency analysis, and a Gibbs sampler algorithm.

In this section we briefly introduce some basic principles of the statistical modelling of regulatory sequences. The general idea is to compare what has been observed in the input sequences with what would have been obtained by having a 'random' data set containing no motif (the 'background' of the sequences against which signals stand out) or a data set built by picking at random some other regulatory sequences (and hence very unlikely to be co-regulated) from the same organism: but a definition of 'randomness' in biological sequences is far from being immediate.

From a theoretical point of view, regulatory regions can be seen as composed of two parts: the binding sites (summed up either with a consensus or with a profile), and 'the rest', which (at least in the process we are studying) does not have any biological role. As we briefly mentioned, the key point for discriminating the signals from the background is *over-representation*: the sequences contain a group of oligos similar to one another that, in the absence of a shared signal, would not be there, at least with the same size and/or the same degree of similarity. Thus, these oligos might play some important role for the function of the sequences: in our case, regulate gene expression.

The first approximation that can be made is to assume that each position of the sequences is independent from the others: thus, the probability of finding a given nucleotide in any position depends

only on the probability with which that nucleotide appears in the sequences, regardless of the nucleotides in adjacent positions or the location of the oligo along the sequence. Thus, the probability of finding a given oligo $p = p_1 \ldots p_m$ in any position of a regulatory sequence can be estimated by the product of the single probabilities of the nucleotides forming it:

$$\Pr(p) = \prod_{i=1}^{m} \Pr(p_i)$$

where $\Pr(p_i)$ is the probability of finding nucleotide p_i in the input sequences, which can in turn be estimated with the observed frequency of p_i in the input data set or in a set of regulatory regions taken from the same organism. Thus, when we represent a motif with a consensus we can estimate its expected frequency according to the probability function just defined, as well as suitable significance measures based on it.[13] For example, one of the most widely used measures is the *z*-score:[14–16]

$$z(p) = \frac{Obs(p) - Exp(p)}{\sqrt{Var(p)}}$$

where $\mathrm{Obs}(p)$ is the number of times consensus p is found (possibly, with substitutions) in the sequences (or, alternatively, the number of sequences p is found in), $\mathrm{Exp}(p)$ and $\mathrm{Var}(p)$ are the expected value and variance of $\mathrm{Obs}(p)$, which can be computed starting from $\Pr(p)$ by using a binomial or a Poisson approximation. Clearly, this measure can take into account how *conserved* the motif is. The more it is conserved, the less alternative forms a motif has, and the lower is its expected frequency.

Likewise, the evaluation the statistical significance of an alignment has to consider simultaneously how much the oligos aligned are similar to one another, and how much the alignment built differs from a random alignment. Let M be a $4 \times m$ profile. Perhaps the most widely used measure is the *information content* (IC, or *relative entropy*) of the profile:

$$IC(M) = \sum_{i=1}^{4} \sum_{j=1}^{m} m_{i,j} \log \frac{m_{i,j}}{b_i}$$

where $m_{i,j}$ is entry in row i and column j of the profile (ranging from 0 to 1), and b_i is the expected frequency of nucleotide i, computed as in the case of consensuses. Clearly, for each column j we have that , and also . It can be seen how this measure accounts for how much each column of the profile is conserved ($m_{i,j}$), and how much the values of the profile differ from what would be expected by aligning random oligos taken from the same sequences (the log ratio). The maximum IC value is obtained when in each position occurs exclusively the rarest nucleotide, while the minimum (0) is reached when in each column the nucleotide frequency equals the background frequency (and thus the profile evaluated is what we should obtain by aligning random oligos). This measure is suitable for comparing alignments built using the same number of sequences. In order to compare alignments of different numbers of oligos, the IC of each one can be multiplied by n, the number of fragments used, yielding the *maximum a posteriori* (MAP, or log-likelihood ratio) score:

$$MAP(M) = -\sum_{i=1}^{4} \sum_{j=1}^{m} n_{i,j} \log \frac{m_{i,j}}{b_i}$$

where $n_{i,j} = n \times m_{i,j}$. Notice that also in this case the different positions in the signal and in the background are assumed to be independent from each other. A similar idea (profile *v*. background) can be used, more simply, to determine whether any oligo p is more likely to be an instance of a motif described by a profile or is part of the background:

$$L(p) = \frac{\Pr(p|M)}{\Pr(p|background)} = \prod_{j=1}^{m} \frac{m_{i,j}}{b_i}$$

where i is the ith base of the oligo. In both cases, the weakest link is the position-independence assumption. If

fact, if we just take every gene upstream region from, say, yeast, count how many times each oligo of a given length appears in the sequences, and compare the result with the expected frequency computed as above (taking into account also the possibility of having in some cases overlapping occurrences of the same oligo) we notice significant differences: thus, the probability of having a given nucleotide in a given position is also influenced by its neighbours, as discussed in Arndt *et al.*,[17] where it is shown that mutation ratios in non-coding DNA depend on the identity of neighbouring bases.

All these considerations have led to the introduction of more sophisticated ways of modelling interdependencies among nucleotides within both binding sites and background regulatory regions. After all, the better the representation of the signal and the background is, the more likely a method is to detect something that significantly differs from the background itself, and the strategy used plays a less essential role. As we have hinted, the position-independent model does not seem to be powerful enough to capture the structure underlying regulatory regions and TFBSs. An improvement, introduced in different recent tools, is to model the background with a higher-order Markov model. Intuitively, when we use a *j*th order Markov model, the probability of finding a nucleotide in a given position of an oligo depends on the *j* nucleotides preceding it in the oligo itself.[18] These parameters can be estimated from the analysis of a number of regulatory regions of different species, leading to organism-specific probability distributions and expected oligo frequencies.

The independence assumption can be relaxed not only for the background, but also in site positions (see Benos *et al.*[19] and Bulyk *et al.*[20] for discussions on this point). In other words, we know that substitutions do not occur independently, and a substitution in a given position might imply another substitution, in a different position, in order to have a given oligo to remain bound by the same TF. Evidence to this conjecture is brought also by the high number of false positives that result from scanning regulatory sequences using position-independent profiles built with known TFBSs.[5] Modelling dependencies among different positions (which might not be consecutive, differently from the Markov models generally used for the background) is quite tricky just for known instances of binding sites,[21,22] since they are likely to change in the different structural classes of transcription factors.[23] However, several observations suggest that at least adjacent positions in binding sites are correlated, that is, the nucleotide in any position *j* influences the choice of nucleotides for positions $j - 1$ and $j + 1$, a fact that can be easily incorporated in any alignment-based algorithm.[20] Another idea is to use a *mixture* of profiles, which describe TFBSs also according to their specificity.[24,25] In other words, binding sites are described with a set of profiles (instead of one), and each of the profiles is used to model a subset of TFBSs with a weight associated with it. The higher is the weight, the more specific the profile is for that TF. The score of an oligo with respect to the profiles is given by the weighted sum of the single scores.

Input parameters

Before delving into the description and discussion of the different methods and approaches, another fundamental issue must be mentioned: how many, and which, parameters the user is supposed to know in advance in order to use an algorithm? Clearly, when fewer parameters are needed, the less prior knowledge is required about the (alleged) common TF and its binding sites. Usually, the methods require as input the *length* of the motif (sometimes also called *width*) or a range for it, especially those that are 'general-purpose' methods not necessarily tailored for TFBSs. Of course, suitable values in this case range from 6–8 to 14–16 nucleotides, and some algorithms just

take these values by default. Then, each method follows a different strategy. Some are just iterated over different motif lengths, and the results of each iteration is output separately (unless otherwise specified, we assume that this is the usual behaviour of the programs we will describe in the following). Others, instead, try to compare and merge the results so to provide the user with a single overall output. Also, another issue is how many of the input sequences are supposed to be co-regulated and thus to share a motif. In their basic version, consensus-based methods require a parameter called *quorum*, that is, a threshold denoting in how many sequences a motif should appear. However, this parameter can be estimated by the algorithm itself according to the input sequences and the statistical measures used to evaluate the output. A similar parameter, denoting how many oligos should be selected from each sequence is anyway needed by alignment based methods (usually the choice is can be 'exactly one', 'zero or one', and, in some cases, 'zero or one or more than one'). Also, consensus-based methods, as mentioned, require a degree of approximation to be specified, that is, how many substitutions are allowed in the occurrences of a motif (or, alternatively, a maximum number of ambiguous IUPAC–IUB symbols allowed in the description of the consensuses) that have to be considered 'valid'. In this case, there are some 'rules of the thumb' that seem to be suitable. For example, as reported in Narasimhan *et al.*,[26] yeast sites usually present two variations in 8-mers, three in motifs 10 nucleotides long, four in motifs of length 12, and so on.

DISCOVERING MOTIFS

Consensus-based methods

The consensus for a set of TFBSs can be seen as a 'perfect' form recognised by a TF. Thus, the idea is to consider all the oligos that differ from a given consensus in no more than *e* positions as belonging to the same group, ie to be binding sites for the same TF. The number of substitutions allowed should in turn depend on the length of the consensus. The algorithmic strategies for consensus-based motifs are mainly based on the following steps. Suppose we know in advance the length *m* of the motif to be found, and are given as input a set of regulatory sequences.

- Enumerate all the possible oligos of length *m*. Each one represents a candidate motif consensus. For each one, count how many times it appears in the sequences (and/or in how many sequences it appears) with no more than *e* substitutions.
- Save all the motifs that appear in all (or most of) the sequences of the set.
- Rank the motifs found according to some statistical measure, and report the highest-ranking motifs.

Quite naturally, if the length *m* is not known in advance, different values have to be tried. This is essentially the first approach introduced to the problem, starting from the mid-1980s.[27–29] However, methods of this kind have been considered for a long time to be 'too slow'. This bad reputation derives mainly from the fact that, given length *m*, there are 4^m candidate consensuses to evaluate, with an exponential growth of the execution time on the motif length. On the other hand, when working on TFBSs the length is never too large (it seldom exceeds 12 or 14 nts); the exhaustive search can be significantly accelerated by organising the input sequences in a suitable indexing structure, such as the suffix tree,[30–33] that yields an execution time exponential in the number of substitutions allowed only (that in turn seldom exceeds four or five); the initial set of candidates of exponential size can be downsized in different ways. All these considerations have led to a rediscovery of this kind of approach in recent years, both in genome-wide scans and in set-specific algorithms.

Clearly, if only exact oligos are considered, that is, no substitutions are allowed in the instances of the same motif, the problem becomes much simpler and its complexity is just linear in the length of the input. Given its computational efficiency this strategy can be employed in genome-wide analyses of over-represented motifs, as for example (among many others) in van Helden[34] for the yeast genome. Here, all sixmers were considered, and their frequency of occurrence in the upstream regions of the genome was compared to an expected value derived from their overall frequency in the non-coding regions of the genome. Similar over-represented oligos can anyway be clustered in a post-processing stage, and considered different forms of binding sites for the same TF.

Also the Verbumculus algorithm[30] considers only exact occurrences. The number of times each oligo is found in the sequences is compared with an expected value based on the nucleotide frequencies in the input, used to estimate the probability of occurrence of nucleotides. A further improvement also considers higher-order background models.[15] By indexing the sequences with a suffix tree, the algorithm reaches a linear time complexity.

Mutations are instead allowed by the SMILE[31] and Weeder[32] algorithms. Also in this case, the exhaustive search for the exponential number of candidate consensi is implemented with the preliminary construction of a suffix tree. While the structure underlying the algorithm is virtually the same, the two approaches differ in how the significance of the motifs found is evaluated. SMILE compares the number of occurrences of a given motif with its occurrences in a random set of sequences of the same size built with a Markov model of any order (which can be chosen by the user) whose parameters are estimated from the input. Alternatively, the user can input another *negative* set of sequences that *should not* contain any instance of the binding sites supposedly appearing in the positive set, used to estimate the most significant motifs found. Clearly, the highest-scoring motifs will be the ones that present the most significant variation between the number of occurrences in the input set and in the random or negative sets. The current version of SMILE requires inputting the motif length(s), the number of substitutions and the quorum value.

The Web implementation of Weeder, instead, directly compares the observed occurrences of an oligo (or a group of oligos) with expected frequencies derived from the oligo-frequency analysis of all the regulatory regions of the same organism of the input sequences. The final score is composed by a general term and a sequence-specific term based respectively on how many sequences each motif appears in and how much conserved it is in each sequence. Different combinations of 'canonical' motif parameters (length, number of substitutions and quorum) are automatically tried by the algorithm in different runs. The interface also analyses and compares the top-scoring motifs of each run in order to detect which ones could be more 'interesting' (even in the case a motif is not the highest scoring one of its run) providing the user with an overall summary and comparison of the results. Finally, the best instances of each motif are selected from the sequences by using a profile built with the oligos found by the consensus-based algorithm, so to possibly include also oligos that present a number of substitutions that exceeds the predefined threshold and to have a more fine-grained ranking of the oligos that fit the substitution threshold.

The YMF algorithm[35,36] is based on 'approximate' consensuses. Other than using ambiguous IUPAC–IUB symbols in the definition, the algorithm also permits explicit searches for motifs composed by two conserved parts separated by a non-conserved region, like the GAL4 signal described in a previous section. The expected number of occurrences of each oligo is estimated with a Markov model of 4th order, and

the significance of each motif is evaluated with a statistical z-score augmented with a term that depends on how many sequences a motif appears in, so to avoid having repeated oligos in a single sequence being reported as highest-scoring motifs (as a consequence, no explicit quorum is needed). As input parameters, the program requires the motif length and a maximum number of approximate IUPAC characters allowed in the definition of consensuses.

A genome-wide method based on consensuses augmented with 'N' symbols is also presented in Vilo *et al.*[37] Motifs are searched in the regulatory regions of yeast genes, pre-clustered according to expression levels derived from microarray experiments. Finally, MobyDick[38] is a tool that permits genome-wide analyses for over-expressed oligos whose description can be augmented with IUPAC ambiguous symbols.

Alignment-based methods

As briefly mentioned before, consensus-based methods have been considered, for quite a long time, unsuitable for the problem. This fact, together with the opinion that consensuses were not flexible enough to describe motifs,[10–12,39] has led to the introduction of a completely different approach. The idea is to build solutions by picking some oligos from the sequences and aligning them in the corresponding profile. Alignments usually do not allow gaps, that is, the oligos must be of the same size. The motifs reported will be those described by the best (highest-scoring) alignments, and the oligos building (or better fitting) each alignment will be considered possible binding sites for the same TF. In this way, the number of parameters needed is reduced mainly to just the motif length, with no need to specify the degree of approximation allowed or a quorum value.

On the other hand, given k input sequences of length n, there are about n^k possible combinations of oligos to be evaluated, regardless of the motif length chosen. From a theoretical point of view, it has been proven that finding the best profile is a NP-hard problem:[40] in practice this means that, whatever the score used, evaluating all the possible profiles is not computationally feasible. Thus, methods that look for the best alignment have to rely on some *heuristic*, that is, some way to prune the search space, avoiding the enumeration of all the possible oligo combinations and building only those alignments that according to some principle (the heuristic) seem to be more likely to be the best ones. While in this way a significant amount of time can be saved, the obvious downside is that the solutions reported cannot be guaranteed to be optimal, but are just the best ones among those considered by the algorithm. In the following we will introduce the algorithms assuming that they look for exactly one site instance per input sequence. All of them, however, can be run also in the so called 'zoops' mode, meaning that each sequence can contain either zero or one motif instance, or in 'zero, one or more than one' mode.

Consensus

Even if the name might be a little deceptive, Consensus is an alignment-based method that employs a *greedy* heuristic.[41,42] Given as input a set of sequences $S_1 \ldots S_k$ the basic version of the algorithm requires as input the length m of the motif to be found, and assumes that the latter occurs once in each sequence. The steps performed by the first version of the algorithm can be summed up as follows:

- All the length m oligos of S_1 are compared with the oligos of length m of S_2. Each comparison produces a $4 \times m$ profile M. Each profile is scored according to its IC, and the highest scoring matrices are saved.

- Each oligo of length m of sequence S_3 is aligned with the matrices saved at the first step, generating a new set of three-sequence profiles; each one is

scored as in the previous step, and the highest scoring ones are saved.

- The second step is repeated for each sequence of the set; the final profiles, output by the program, will contain one oligo for each input sequence.

The algorithm is *greedy*, that is, at each step saves the best partial alignments only, hoping that they will eventually lead to the optimal one. Obviously, the more conserved the motif is, the more likely is the algorithm to find it. Otherwise, the risk is to store in the first steps matrices corresponding to random (but similar enough) oligos, and to discard the one that would have led to the highest-scoring solution. Further improvements are presented in the WConsensus algorithm.[42] They include the possibility of finding motifs that do not occur or appear more than once in each sequence, and avoid explicitly requiring the length parameter from the user. Moreover, profiles are built by comparing directly all pairs of sequences, and hence the problem of the result depending on the order of sequences is avoided. Also, the calculation of a *p-value* for an alignment is introduced. The *p*-value gives an estimate of the probability of finding a profile with the same IC score by chance, which is especially useful in comparing alignments with different lengths and different numbers of sites, cases where comparisons based on IC alone are not sufficient.

MEME

Another way of looking at the problem of finding the best alignment profile is to 'guess' the position in the input sequences of the regions forming it. Given a profile *M*, the MEME (Multiple Expectation Maximisation for Motif Elicitation) algorithm[43] evaluates the likelihood of each sequence region of a length *m* to fit the profile with respect to the background of the sequences, while the rest of the sequence should fit the background better than the profile. According to this principle, a likelihood value $z_{i,j}$ (normalised such that the sum over all the $z_{i,j}$ values of sequence *j* equals 1) is computed for each position *i* of each input sequence *j*. This is the E (Expectation) step. Then, the algorithm builds a new alignment profile by putting together all the sequence regions of length *m*, but weighing each one with the corresponding $z_{i,j}$ value. This is the M (Maximisation) step. The algorithm starts by building a different profile from each *m*-mer in the input sequences, using a frequency value of $\frac{1}{2}$ for the nucleotides of the oligo and 1/6 for the others. Then, for each profile (each *m*-mer in the input) it performs a single E and a single M step. The highest MAP scoring profile obtained (after the single iteration) is further optimised with additional EM steps, until no further increase on the score is obtained. Finally, the profile is reported, and its oligos are removed from the input sequences. Then, the algorithm is restarted, until a number of profiles that can be specified as input has been generated. Thus, MEME can detect multiple motifs within the same set of sequences within a single run.

The Gibbs samplers

One of the most successful approaches to the problem, for the part concerning the heuristic used to find the highest-scoring profiles, has been the Gibbs sampling strategy, first introduced for motif discovery in protein sequences in Lawrence *et al.*[44] and Neuwald *et al.*[45] but nevertheless perfectly suitable also for nucleotide sequences (and recently further fine-tuned to TFBSs). The best measure of its success is perhaps the number of times it has been used in the algorithmic part of different methods, which varied the statistical measures used to generate and evaluate the results. The main motivation was to improve a EM local search strategy[46] (similar to the one employed by MEME), so to avoid the problem of premature convergence to local maxima of the IC and MAP functions. The basic idea, designed for sequence sets with exactly one site

instance per sequence can be summarised as follows:

- An oligo of length m is chosen at random in each of the k input sequences (at the beginning, with uniform probability).
- One of the k sequences is chosen at random: let S be this sequence.
- A $4 \times m$ profile M is built with the oligos that had been selected in the other $k - 1$ sequences.
- For each position i in S, let $p_i =$ the m-mer of S starting at position i. For each p_i a likelihood value $L(p_i)$ is computed, representing how well it p_i fits the model induced by the matrix M with respect to the background nucleotide distribution.
- A new probability value, proportional to $L(p_i)$, is assigned to each position i of S. Thus, the oligos that fit well in the alignment described by M are more likely to be chosen at the next cycle.
- Go to the first step: now the probability with which the m-mers of sequence S can be picked are those computed at the previous step.

These steps are iterated a number of times, or until convergence is reached. This variant of the algorithm is also known as the *site sampler*. The main difference with MEME is in the first step: while the local search always picks oligos deterministically according to how much they fit a profile, the Gibbs sampler chooses the fragment that has to be added to the profile in a stochastic way. At the beginning all the oligos have the same probability of being chosen; in successive iterations, those that better fit the profile are more likely (but not certain) to be selected. The algorithm is thus less likely to get stuck in local optima; on the other hand, given its probabilistic nature, it has often to be run different times, and the final results can be obtained by comparing the outputs of each run. Additions to the basic algorithm were presented successively,[45] allowing multiple occurrences of a motif within the same sequence, or, conversely, the motif did not have to occur in every sequence (algorithm known as *motif sampler*). This variant, however, needs an estimate of the overall number of times a motif is expected to appear in the input sequences. Modifications of the basic Gibbs sampling technique especially devised for DNA sequences are described in Hughes *et al.*[47] and Workman and Stormo.[48] AlignACE[47] is a program where the basic Gibbs sampling algorithm is fine-tuned in order to work on DNA regulatory sequences, including for example both strands of each input sequence and introducing a different sampling technique that also considers similarity in the position relative to the TSS of each of the oligos of a group. That is, a functional motif should correspond to similar regions appearing at similar distance from the TSS. In the ANN-Spec algorithm,[48] a Gibbs sampling method is combined with an artificial neural network that replaces the frequency matrix. Instead of aligning the oligos selected and scoring the matrix, the algorithm trains a neural network in order to recognise the oligos selected against the rest of the sequences.

HOW TO USE A MOTIF DISCOVERY TOOL

There are several examples on how motif discovery methods can be efficiently used (integrated with other tools) for to obtain meaningful results and insight. Among many others, we might also cite Lee *et al.*,[49] Ohler *et al.*,[50] Rajewsky *et al.*[51] and Beer and Tavazoie.[52] The most recent study we found is presented in Beer and Tavazoie.[52] First of all, yeast genes are clustered according to a large number of expression values (255 in all) obtained from several experiments whose data are publicly available. In this way, the

clustering phase is more reliable and fine-grained than in experiments where fewer values are used. Each of the 40 clusters obtained is then processed separately, by examining the upstream regions of the genes with AlignAce (as well as by detecting the presence in them of known TFBSs listed in the SCPD database[7]). At this point, as usual a large number of candidate motifs are obtained. But, these motifs are further examined, in order to find correlations in relative position, distance from the TSS, and strand orientation. In this way, a subset of 'interesting' motifs is selected, as well as information on which motifs might cooperate to regulate gene expression and with what effect. Finally, a further validation is performed, in order to determine to which extent the motifs found can represent real TFBSs. In this last phase, the analysis outlined at the previous points is performed by leaving a subset of the genes out (the *test set*). Motifs are collected from other genes and their expression values. Then, the motifs found are used to *predict* the expression values of the genes left in the test set (and not used to build motifs), and the predicted expression values are compared to the known values. This analysis is repeated by using different test sets with a cross-validation technique, and a similar example is presented for a subset of *Caenorhabditis elegans* genes.

All in all, the results reported are very encouraging, with a significant correlation between real expression values and those predicted by using solely the motifs detected. This study proves without any doubt that, while we are still far from the 'crystal ball' mentioned in the introduction (and that many researchers expect tools to be), motif discovery can provide very useful and meaningful results, if used with a grain of salt.

Improvements

Historically speaking, the algorithms we have described were the first alignment-based methods to be introduced, and as we have seen they are still widely used today with good results. In any case, their heuristic (in how solutions are generated) and probabilistic (in how solutions are evaluated) nature lends itself to different improvements. Current directions of research are mainly the following:

- Improving the heuristics: the algorithms happen to miss a motif altogether because they do not include the optimal matrix (corresponding to the motif) among the candidate solutions. The reason could also lie in the choice of initial profiles that are optimised.

- Improving the scoring function: heuristics work just fine, but the algorithms happen to miss a motif because if we use the traditional IC and MAP scores, the corresponding frequency matrix is not the highest-scoring one (or, alternatively, the real motif is 'lost' among many random motifs with higher or similar scores, and further work is needed to discriminate it).

- Looking for a single motif in some cases is not enough: that is, a motif is made of two or more TFBS, located within short distance from each other, whose biological function (and statistical relevance) is the effect of their simultaneous appearance within a promoter, and their relative distance.[53] Moreover, each of the cooperating TFBSs is not overrepresented enough to be detected by itself (as instead we saw in the example we presented). These have also been called *composite* or *structured* or (in case of pairs) *dyad* motifs.

The main motivation for the first point was a series of tests performed on artificially generated data sets. In these tests, a 'simulated' motif was implanted in a set random sequences with uniform nucleotide composition.[48,54] The length of the motif varied from 8 to 20 nucleotides, with a proportionally

increasing number of mutations that could occur in any position of the motif. The three traditional alignment-based methods did not perform very well, since they got stuck into random (and lower scoring) alignments. For example, they performed poorly when a 15 nucleotide long motif was implanted in 20 sequences of 600 bp, with four substitutions in each occurrence (that could be at any position), even if the correct motif length was given as input and, on the other hand, the artificial motif was the highest-scoring one. This observation led to the introduction of a number of new approaches and heuristics to the problem. Among others, SP-Star and Winnower,[54] Projection,[55] MITRA,[56] Multiprofiler[57] and P-Branching[58] may be mentioned. While aimed at finding the highest-scoring profile, these methods (that essentially differ in how initial profiles are built before an optimisation procedure) nevertheless require as input parameters the length of the motif and the number of mutations allowed for its occurrences, as consensus-based methods that can guarantee to find the optimal solution in acceptable time.[32] All in all, the results on this line, and on the artificial data sets, are quite satisfactory: unfortunately, in many cases the best solution is not 'unique': that is, it is not the only oligo of length 15 that appears in all the sequences with four mutations (and thus, basically, the only motif that satisfies some properties), but it is one of the thousands of oligos of, say, length 8 appearing in the sequences with two mutations. And the issue of whether IC and MAP scores are suitable for finding real binding sites remains open.

On the other hand, research moved along the other direction, that is, finding a scoring function able to reflect, as much as possible, the 'biological' side of TFBSs (and thus solving the one-out-of-thousands issue). As we have discussed, one major point was (and still is), how to model 'randomness', that is, what happens in the *absence* of shared TFBSs, or, in other words, in the 'background' of the sequences.

The MIRA algorithm[26] implements more involved models for both signals and background. It is essentially a local search alignment-based method, where starting alignments are built by picking, for each oligo appearing in the sequences, all the other oligonucleotides differing from it in a fixed number of positions, with a strategy analogous to consensus-based methods. Each profile is scored with a modified version of the IC function, which takes into account both correlations between adjacent nucleotides in the motif, and uses a background model of order four.

Also the MEME algorithm has been improved and enhanced over the years. The current version (3.0) uses a higher-order background model, accepts a range value for motif lengths, and post-processes automatically the results of different lengths producing a single overall output.

Enhanced Gibbs samplers

In this section we separately list some of the newest algorithms, which are still based on the Gibbs sampling technique but present some improvements over the 'traditional' method, essentially based on the considerations we outlined in the previous section.

Bioprospector[59] is a Gibbs sampler with some additional features. First of all, the background is described with a third order Markov model based on the genome-wide analysis of different organisms (thus users must specify which species their sequences are taken from). Also, the choice of the sequence to be sampled is modified in order to take into account the fact that a sequence can contain zero or more than one instance of a motif. Finally, it can detect also composite (as we will discuss in the following) motifs.

Another version of the Gibbs motif sampler is presented in Thijs *et al.*,[60] and is part of a larger toolset for the analysis of regulatory sequences.[61] Also in this case, additional features are the possibility of considering multiple occurrences of the same motif within the same sequence, as

often is the case in eukaryotic promoters (or zero, as well), and a higher-order organism-specific modeling of the background.

GLAM[62] is a Gibbs sampling algorithm especially tailored to TFBSs, where the sampling procedure as well as the IC score have been modified in order to compare profiles of different size sparing the user the visual inspection and comparison of the results obtained on different lengths. The optimal motif length is computed with a simulated annealing strategy.

Also the authors of the original Gibbs sampler have released a version of their algorithm designed and adapted for TFBSs,[63] with a new method for the modelling of the sequence background. The idea is to use position-specific frequencies for the bases: in other words, if an oligo is located 100 bps upstream of the transcription start site, its expected frequency is estimated by analysing the oligos that appear at approximately the same distance from the TSS with a Bayesian segmentation algorithm. As a matter of fact, different analyses have shown strong preferences of the sites for given position ranges with respect to the TSS,[64] while the same oligo seems to be inactive when appearing at the 'wrong' place. Thus, the same oligos can yield profiles of completely different significance values when located at different positions relative to the TSS. This new version is also able to search simultaneously for multiple motifs within the same data set (instead of optimising a single motif), and sets all the parameters needed by the algorithm to default values suitable for TFBSs.

Finding composite motifs

Sometimes, a motif cannot be detected because its presence alone is not significant enough or, in other words, it does not act alone in the regulation of gene expression but appears to be correlated with other motifs. Thus, the resulting biological activity is the effect of the simultaneous presence of different sites, as well as their order, orientation, strand and relative position.[53,65] Some of the methods we mentioned include explicitly the possibility of detecting these combinations of single motifs, instead of leaving this task to a post-processing stage. Clearly, suitable statistical modelling has to be defined also for this case.[66] Among consensus-based methods, SMILE can discover motifs made of an arbitrary number of separate parts, provided with their number, respective size and maximum relative distance within the sequences. A similar strategy is employed by MITRA, which can detect *dyad* motifs composed of two parts. Among Gibbs samplers, dyad motifs are also included in Bioprospector, which needs the size of each part and a gap range for the region separating them. Mermaid[67] implements a variation of the local search of MEME, with some changes in the IC function used to score profiles, also allowing motifs composed of any number of parts. Finally, the ANN-Spec approach, based on neural networks, is extended to finding binding sites for cooperating TFs (with parameters analogous to the other tools) in the Co-Bind algorithm.[68] The main drawback of these methods usually is that the complexity is increases with the width of the gap-separating motifs, and the latter parameter is very often unknown and hard to estimate.

CONSENSUS OR ALIGNMENT?

The debate on which 'philosophy' is more suitable for capturing TFBSs is nearly as old as the problem itself.[10,11,39] Clearly, profiles (and derivates) are more flexible, and seem to require less prior knowledge to be discovered. On the other hand, as we have seen in the previous section, the failure of an alignment-based method can derive from many factors: for example, the methods always report some signal, but the user has to figure out whether it is a real signal, and if it is not whether it is because of the absence of a common TF or because the method simply did not see another more interesting motif (for this task, most

alignment-based algorithms output also a *p*-value associated with each profile, expressing the probability of obtaining such an alignment by chance, or perform other analogous statistical tests).

An interesting comparison was presented in Sinha and Tompa.[69] The contestants were YMF, MEME (version 3.0) and AlignACE. The battlefield was the set of TFs listed in the yeast promoter database (SCPD).[7] A different data set was built for each TF, composed of all the promoter sequences listed in the database as containing an experimentally known binding site for the TF (thus, each data set was 'clean', ie the motif appeared in all the sequences, and each data set was known to contain a motif that could appear more than once in each sequence). In this way 34 data sets were built, containing a varying number of sequences of 800 bp. The performance of the algorithms was measured as how much the predicted binding sites matched the solution. Rather than showing a definite preference for one of the methods, the results highlighted that the two approaches seem to be 'complementary'. In some cases, most of the correct site instances were picked (or, for different reasons, completely missed) by all the three methods; but on some data sets, the motif seemed to fit the consensus description much better than the profile one, and vice versa in some others.

The results of this test just seem to hint that there cannot be a definite preference for either way of modelling TFBSs, especially in the absence of any prior information concerning the structure and the degree of conservation of a motif. Rather, the lesson we might learn is that, given a set of sequences, the best way to proceed is to try different methods, based on wholly different principles, and to compare their results. First of all, once we have the outputs we should check whether one or more programs found in the sequences something that has already been discovered and characterised, by checking databases such as TRANSFAC[9] or other species-specific data repositories. Otherwise, if the methods we used agree on some novel motif (even if the overall representation is different, all the programs usually output explicitly the sites used to construct the best solutions, and therefore the results can be directly compared), then the motif can be considered a very promising candidate for having a true biological activity. Also, virtually all the methods we mentioned in this paper output more than a motif, and sometimes the 'real' one is not the highest scoring one, but it appears in the top-five or top-ten list. Or, in other cases, they output separate lists of motifs of different length, making hard to the user to figure out which motif of which length should be considered more reliable. Comparing the results of different methods might be of help also in this case, that is, if a motif is caught among the highest ranking ones (but not necessarily the best one) by more than one method, then it might be worth further investigation. When the methods totally disagree, the issue becomes more complicated, since the significance measures differ from algorithm to algorithm, and it is very hard to judge whether the highest scoring consensus of, say, Weeder or YMF is better or worse than the highest-scoring profile of MEME or the Gibbs sampler, and if one of them (or none) actually corresponds to real conserved TFBSs. In this case, however, there is the latest resort: the experience and the judgment of the biologist using the tools that will never be replaced by any computer program.

THE NEXT STEP: BACK TO SQUARE ONE?

Throughout the paper, methods, approaches and philosophies have been described that assume (as the developers of the methods often did) that the user has built a feasible input data set, that is, all or most of the genes he/she put together are actually co-regulated. While this is surely the case of the data sets used to evaluate and compare the performance of the different programs, it is much less likely in real life. That is, users often just suspect

(or maybe hope) that their sequences are co-regulated. As we have seen, most of the methods can somehow accommodate the fact that the common TF regulates only a subset of the genes, by setting a suitable quorum threshold or using the 'zero or one occurrence' mode. But, the likelihood of picking the correct motif clearly decreases as the number of spurious sequences in the data set is increased, since the basic feature all the methods hinge upon, over-representation, starts to fade into background noise. However, in some cases the input data set is derived from some preliminary analysis of other data, such as microarray expression levels: that is, regulatory regions of genes belonging to the same cluster, obtained with some method (see, for example, Huber *et al.*[70] and Baldi and Wesley Hatfield[71]) are in turn fed to some motif-discovery program as we previously described and shown in, among many others, Vilo *et al.*[37] and Tavazoie *et al.*[72]

The large and constantly increasing number of gene clustering methods based on expression data (whole journal issues solely dedicated to this problem often appear) is a clear sign that also putting together the right genes is an open and far from being solved problem. On the other hand, there is significant evidence that variations in expression levels can be directly correlated with the presence of TFBSs in the upstream regions of a gene,[73–75] and recent research has moved also in this direction. Instead of leaving to the user the task of preparing a set of related sequences, some new methods can work on whole genomes (or large data sets of thousands of genes), taking as input directly the expression levels. The same idea can be used to improve whole-genome analyses. In fact, a given oligo appearing a limited number of times can be completely missed by considering over-representation only: but if its few occurrences happen to be in co-regulated genes (or genes having similar expression profile), then its importance has to be reconsidered. Ideally, integrating sequence and expression data should provide different advantages: first of all, a positive and negative sequence set are supplied implicitly, since a motif should appear in the promoters of over-expressed genes and not in those under-expressed (as in Haverty *et al.*,[76] where TRANSFAC profiles of already known sites are used); then, it bypasses completely the clustering phase and the bias in the data sets that it inevitably causes; finally, we no longer need to have an idea on how many sequences in the data set could contain an instance of a motif. While the feasibility of this approach on human or mouse sequences is still debated (see, for example, Dieterich *et al.*[77]), since in this case the regions to be considered are much larger and more prone to false positives, some preliminary analyses on yeast microarray and ChIP experiments have given encouraging results.

The aim of REDUCE[78,79] is to correlate gene expression levels with the presence of conserved motifs appearing in the regulatory regions. Only exact consensus occurrences are considered. Each oligo is scored according to its occurrence frequency in the input sequences, weighted by the expression level values associated with the sequences it appears in. Then, the expression values associated with the sequences are 'explained' according to the highest-scoring oligos with a linear regression technique.

MDscan[80] directly explores the regulatory regions of the most over-expressed genes (whose number has to be supplied by the user) of a set with a consensus-based strategy, building a frequency matrix from the occurrences collected for each consensus. Then, each matrix is ranked according to its MAP score against a background model of the 3rd order, built from the intergenic regions of yeast (the organism analysed in the article). The highest-scoring matrices are saved and undergo a final local optimisation step that considers a larger set of highly expressed genes (although under-expressed genes are not

simultaneously considered, and thus no negative set is used as counter-example). In Conlon *et al.*,[81] MDscan is integrated with Motif Regressor, which performs a linear regression of microarray expression values (similar to REDUCE) based, instead of single oligos, on the motif profiles reported by MDscan.

Clearly, while this is a new and promising direction for direction, developers of future methods should also bear in mind, and incorporate in their solutions, all the considerations we scattered throughout the paper when describing the philosophy of existing methods and possible improvements (or, better, those considerations they think make sense), as well as all the biological information they can collect from studies on how and why TFBSs influence and modulate gene expression.

CONCLUSIONS

Perhaps as a consequence of the simultaneous explosion of different types of data coming from the world of molecular biology, such as whole genomes, transcriptomes and gene expression analyses, the prediction of TFBSs in regulatory regions of related genes has become one of the hottest topics in bioinformatics, as witnessed by the constant growth of the number of articles and methods devoted to this task. As we have seen, the problem is extremely challenging at every level, that is, from the modelling of binding sites, to the construction of candidate solutions, to the evaluation of the best solutions. Moreover, most of the tests performed so far are based on the results of the analysis of yeast genes, where regulatory regions are short (less than 1000 bp) and simple, while for human much longer regions with a complex organisation of regulatory modules (enhancers, silencers, etc.) have to be considered.[2] This paper has provided a survey of existing algorithms and research trends, covering as many methods and standpoints as possible. Apologies are given in advance for possible unintentional omissions that are by no means due to a negative evaluation from our point. In any case, new approaches and ideas will surely appear even while this article is in press, and we strongly advise the interested reader to check journals and conferences on a regular basis for new updates and methods in one of the fastest-evolving branches of bioinformatics.

Acknowledgments

This work was supported by FIRB project 'Bioinformatica per la Genomica e la Proteomica' (Ministero dell'Istruzione e Ricerca Scientifica, Italy) and by Telethon. We thank David Horner for valuable comments on the manuscript.

References

1. Lemon, B. and Tjian, R. (2000), 'Orchestrated response: A symphony of transcription factors for gene control', *Genes Dev.*, Vol. 14**,** pp. 2551–2569.
2. Levine, M. and Tjian, R. (2003), 'Transcription regulation and animal diversity', *Nature*, Vol. 424, pp. 147–151.
3. Okazaki, Y., Furuno, M., Kasukawa, T. *et al.* (2002), 'Analysis of the mouse transcriptome based on functional annotation of 60,770 full-length cDNAs', *Nature*, Vol. 420, pp. 563–573.
4. Suzuki, Y., Yamashita, R., Sugano, S. and Nakai, K. (2004), 'DBTSS, DataBase of Transcriptional Start Sites: Progress report 2004', *Nucleic Acids Res.*, Vol. 32 Database issue, pp. D78–81.
5. Rahmann, S., Muller, T. and Vingron, M. (2003), 'On the power of profiles for transcription factor binding site detection', *Statist. Appl. Genet. Mol. Biol.*, Vol. 2.
6. Bulyk, M. L. (2003), 'Computational prediction of transcription-factor binding site locations', *Genome Biol.*, Vol. 5, p. 201.
7. Zhu, J. and Zhang, M. Q. (1999), SCPD: A promoter database of the yeast *Saccharomyces cerevisiae*. *Bioinformatics*, Vol. 15, pp. 607–611.
8. Nomenclature Committee of the International Union of Biochemistry (NC-IUB) (1986), 'Nomenclature for incompletely specified bases in nucleic acid sequences. Recommendations 1984', *Proc. Natl Acad. Sci. USA*, Vol. 83, pp. 4–8.
9. Matys, V., Fricke, E., Geffers, R. *et al.* (2003), 'TRANSFAC: Transcriptional regulation, from patterns to profiles', *Nucleic Acids Res.*, Vol. 31, pp. 374–378.
10. Frech, K., Quandt, K. and Werner, T. (1997), 'Software for the analysis of DNA sequence

elements of transcription', *Comput. Appl. Biosci.*, Vol. 13, pp. 89–97.

11. Frech, K., Quandt, K. and Werner, T. (1997), 'Finding protein-binding sites in DNA sequences: The next generation', *Trends Biochem. Sci.*, Vol. 22, pp. 103–104.
12. Stormo, G. D. (2000), 'DNA binding sites: Representation and discovery', *Bioinformatics*, Vol. 16, pp. 16–23.
13. Reinert, G., Schbath, S. and Waterman, M. S. (2000), 'Probabilistic and statistical properties of words: An overview', *J. Comput. Biol.*, Vol. 7, pp. 1–46.
14. Tompa, M. (1999), 'An exact method for finding short motifs in sequences, with application to the ribosome binding site problem', *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, pp. 262–271.
15. Apostolico, A., Bock, M. E. and Lonardi, S. (2003), 'Monotony of surprise and large-scale quest for unusual words', *J. Comput. Biol.*, Vol. 10, pp. 283–311.
16. Leung, M. Y., Marsh, G. M. and Speed, T. P. (1996), 'Over- and underrepresentation of short DNA words in herpesvirus genomes', *J. Comput. Biol.*, Vol. 3, pp. 345–360.
17. Arndt, P. F., Burge, C. B. and Hwa, T. (2003), 'DNA sequence evolution with neighbor-dependent mutation', *J. Comput. Biol.*, Vol. 10, pp. 313–322.
18. Thijs, G., Lescot, M., Marchal, K. *et al.* (2001), 'A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling', *Bioinformatics*, Vol. 17, pp. 1113–1122.
19. Benos, P. V., Bulyk, M. L. and Stormo, G. D. (2002), 'Additivity in protein–DNA interactions: How good an approximation is it?', *Nucleic Acids Res.*, Vol. 30, pp. 4442–4451.
20. Bulyk, M. L., Johnson, P. L. and Church, G. M. (2002), 'Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors', *Nucleic Acids Res.*, Vol. 30, pp. 1255–1261.
21. Ellrott, K., Yang, C., Sladek, F. M. and Jiang, T. (2002), 'Identifying transcription factor binding sites through Markov chain optimization', *Bioinformatics*, Vol. 18, Suppl. 2, pp. S100–S109.
22. Barash, Y., Elidan, G., Friedman, N. and Kaplan, T. (2003), 'Modeling dependencies in protein-DNA binding sites', *RECOMB '03*, ACM, Berlin, Germany, pp. 28–37.
23. Suzuki, M. and Yagi, N. (1994), 'DNA recognition code of transcription factors in the helix-turn-helix, probe helix, hormone receptor, and zinc finger families', *Proc. Natl Acad. Sci. USA*, Vol. 91, pp. 12357–12361.
24. Roulet, E., Busso, S., Camargo, A. A. *et al.* (2002), 'High-throughput SELEX SAGE method for quantitative modeling of transcription-factor binding sites', *Nat. Biotechnol.*, Vol. 20, pp. 831–835.
25. King, O. D. and Roth, F. P. (2003), 'A non-parametric model for transcription factor binding sites', *Nucleic Acids Res.*, Vol. 31, p. e116.
26. Narasimhan, C., LoCascio, P. and Uberbacher, E. (2003), 'Background rareness-based iterative multiple sequence alignment algorithm for regulatory element detection', *Bioinformatics*, Vol. 19, pp. 1952–1963.
27. Galas, D. J., Eggert, M. and Waterman, M. S. (1985), 'Rigorous pattern-recognition methods for DNA sequences. Analysis of promoter sequences from *Escherichia coli*', *J. Mol. Biol.*, Vol. 186, pp. 117–128.
28. Sadler, J. R., Waterman, M. S. and Smith, T. F. (1983), 'Regulatory pattern identification in nucleic acid sequences', *Nucleic Acids Res.*, Vol. 11, pp. 2221–2231.
29. Waterman, M. S., Arratia, R. and Galas, D. J. (1984), 'Pattern recognition in several sequences: Consensus and alignment', *Bull. Math. Biol.*, Vol. 46, pp. 515–527.
30. Apostolico, A., Bock, M. E., Lonardi, S. and Xu, X. (2000), 'Efficient detection of unusual words', *J. Comput. Biol.*, Vol. 7, pp. 71–94.
31. 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.*, Vol. 7, pp. 345–362.
32. Pavesi, G., Mauri, G. and Pesole, G. (2001), 'An algorithm for finding signals of unknown length in DNA sequences', *Bioinformatics*, Vol. 17, Suppl. 1, pp. S207–214.
33. Gusfield, D. (1997), 'Algorithms on Strings, Trees, and Sequences : Computer Science and Computational Biology', Cambridge University Press, Cambridge.
34. van Helden, J. (2003), 'Regulatory sequence analysis tools', *Nucleic Acids Res.*, Vol. 31, pp. 3593–3596.
35. Sinha, S. and Tompa, M. (2002), 'Discovery of novel transcription factor binding sites by statistical overrepresentation', *Nucleic Acids Res.*, Vol. 30, pp. 5549–5560.
36. Sinha, S. and Tompa, M. (2003), 'YMF: A program for discovery of novel transcription factor binding sites by statistical overrepresentation', *Nucleic Acids Res.*, Vol. 31, pp. 3586–3588.
37. Vilo, J., Brazma, A., Jonassen, I. *et al.* (2000), 'Mining for putative regulatory elements in the yeast genome using gene expression data', *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, Vol. 8, pp. 384–394.

38. Bussemaker, H. J., Li, H. and Siggia, E. D. (2000), 'Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis', *Proc. Natl Acad. Sci. USA*, Vol. 97, pp. 10096–10100.

39. Berg, O. G. and von Hippel, P. H. (1988), 'Selection of DNA binding sites by regulatory proteins', *Trends Biochem. Sci.*, Vol. 13, pp. 207–211.

40. Akutsu, T., Arimura, H. and Shimozono, S. (2000), 'On approximation algorithms for local multiple alignment', in 'RECOMB 2000', ACM, Tokyo, pp. 1–12.

41. Hertz, G. Z., Hartzell, G. W. and Stormo, G. D. (1990), 'Identification of consensus patterns in unaligned DNA sequences known to be functionally related', *Comput. Appl. Biosci.*, Vol. 6, pp. 81–92.

42. Hertz, G. Z. and Stormo, G. D. (1999), 'Identifying DNA and protein patterns with statistically significant alignments of multiple sequences', *Bioinformatics*, Vol. 15, pp. 563–577.

43. Bailey, T. L. and Elkan, C. (1994), 'Fitting a mixture model by expectation maximization to discover motifs in biopolymers', *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, Vol. 2, pp. 28–36.

44. Lawrence, C. E., Altschul, S. F., Boguski, M. S. *et al.* (1993), 'Detecting subtle sequence signals: A Gibbs sampling strategy for multiple alignment', *Science*, Vol. 262, pp. 208–214.

45. Neuwald, A. F., Liu, J. S. and Lawrence, C. E. (1995), 'Gibbs motif sampling: Detection of bacterial outer membrane protein repeats', *Protein Sci.*, Vol. 4, pp. 1618–1632.

46. Lawrence, C. E. and Reilly, A. A. (1990), 'An expectation maximization (EM), algorithm for the identification and characterization of common sites in unaligned biopolymer sequences', *Proteins*, Vol. 7, pp. 41–51.

47. Hughes, J. D., Estep, P. W., Tavazoie, S. and Church, G. M. (2000), 'Computational identification of *cis*-regulatory elements associated with groups of functionally related genes in *Saccharomyces cerevisiae*', *J. Mol. Biol.*, Vol. 296, pp. 1205–1214.

48. Workman, C. T. and Stormo, G. D. (2000), 'ANN-Spec: A method for discovering transcription factor binding sites with improved specificity', *Pacific Symp. Biocomput.*, pp. 467–478.

49. Lee, T. I., Rinaldi, N. J., Robert, F. *et al.* (2002), 'Transcriptional regulatory networks in *Saccharomyces cerevisiae*', *Science*, Vol. 298, pp. 799–804.

50. Ohler, U., Liao, G. C., Niemann, H. and Rubin, G. M. (2002), 'Computational analysis of core promoters in the *Drosophila* genome', *Genome Biol.*, Vol. 3, pp. ??

51. Rajewsky, N., Vergassola, M., Gaul, U. and Siggia, E. D. (2002), 'Computational detection of genomic *cis*-regulatory modules applied to body patterning in the early *Drosophila* embryo', *BMC Bioinformatics*, Vol. 3, p. 30.

52. Beer, M. A. and Tavazoie, S. (2004), 'Predicting gene expression from sequence', *Cell*, Vol. 117, pp. 185–198.

53. Chiang, D. Y., Moses, A. M., Kellis, M. *et al.* (2003), 'Phylogenetically and spatially conserved word pairs associated with gene-expression changes in yeasts', *Genome Biol.*, Vol. 4, p. R43.

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

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

56. Eskin, E. and Pevzner, P. A. (2002), 'Finding composite regulatory patterns in DNA sequences', *Bioinformatics*, Vol. 18, Suppl. 1, pp. S354–363.

57. Keich, U. and Pevzner, P. A. (2002), 'Finding motifs in the twilight zone', *Bioinformatics*, Vol. 18, pp. 1374–1381.

58. Price, A., Ramabhadran, S. and Pevzner, P. A. (2003), 'Finding subtle motifs by branching from sample strings', *Bioinformatics*, Vol. 19, Suppl. 2, pp. II149–II155.

58. Liu, X., Brutlag, D. L. and Liu, J. S. (2001), 'BioProspector: Discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes', *Pacific Symp. Biocomput.*, pp. 127–138.

60. Thijs, G., Marchal, K., Lescot, M. *et al.* (2002), 'A Gibbs sampling method to detect overrepresented motifs in the upstream regions of coexpressed genes', *J. Comput. Biol.*, Vol. 9, pp. 447–464.

61. Aerts, S., Thijs, G., Coessens, B. *et al.* (2003), 'Toucan: Deciphering the *cis*-regulatory logic of coregulated genes', *Nucleic Acids Res.*, Vol. 31, pp. 1753–1764.

62. Frith, M. C., Hansen, U., Spouge, J. L. and Weng, Z. (2004), 'Finding functional sequence elements by multiple local alignment', *Nucleic Acids Res.*, Vol. 32, pp. 189–200.

63. Thompson, W., Rouchka, E. C. and Lawrence, C. E. (2003), 'Gibbs Recursive Sampler: Finding transcription factor binding sites', *Nucleic Acids Res.*, Vol. 31, pp. 3580–3585.

64. Hampson, S., Kibler, D. and Baldi, P. (2002), 'Distribution patterns of over-represented *k*-mers in non-coding yeast DNA', *Bioinformatics*, Vol. 18, pp. 513–528.

65. Makeev, V. J., Lifanov, A. P., Nazina, A. G. and Papatsenko, D. A. (2003), 'Distance preferences in the arrangement of binding

motifs and hierarchical levels in organization of transcription regulatory information', *Nucleic Acids Res.*, Vol. 31, pp. 6016–6026.

66. Robin, S., Daudin, J. J., Richard, H. *et al.* (2002), 'Occurrence probability of structured motifs in random sequences', *J. Comput. Biol.*, Vol. 9, pp. 761–773.
67. Hu, Y. J. (2003), 'Finding subtle motifs with variable gaps in unaligned DNA sequences', *Comput. Methods Programs Biomed.*, Vol. 70, pp. 11–20.
68. GuhaThakurta, D. and Stormo, G. D. (2001), 'Identifying target sites for cooperatively binding factors', *Bioinformatics*, Vol. 17, pp. 608–621.
69. Sinha, S. and Tompa, M. (2003), 'Performance comparison of algorithms for finding transcription factor binding sites, in 'Third IEEE Symposium on Bioinformatics and Bioengineering', Washington, DC, pp. 69–78.
70. Huber, W., Heydebreck, A. and Vingron, M. (2003), 'Analysis of microarray gene expression data', in 'Handbook of Statistical Genetics', 2nd edn, Wiley.
71. Baldi, P. and Wesley Hatfield, G. (2002), 'DNA Microarrays and Gene Regulation', Cambridge University Press.
72. Tavazoie, S., Hughes, J. D., Campbell, M. J. *et al.* (1999), 'Systematic determination of genetic network architecture', *Nat. Genet.*, Vol. 22, pp. 281–285.
73. Palin, K., Ukkonen, E., Brazma, A. and Vilo, J. (2002), 'Correlating gene promoters and expression in gene disruption experiments', *Bioinformatics*, Vol. 18, Suppl. 2, pp. S172–180.
74. Chiang, D. Y., Brown, P. O. and Eisen, M. B. (2001), 'Visualizing associations between genome sequences and gene expression data using genome-mean expression profiles', *Bioinformatics*, Vol. 17, Suppl. 1, pp. S49–55.
75. Clarke, N. D. and Granek, J. A. (2003), 'Rank order metrics for quantifying the association of sequence features with gene regulation', *Bioinformatics*, Vol. 19, pp. 212–218.
76. Haverty, P. M., Hansen, U. and Weng, Z. (2004), 'Computational inference of transcriptional regulatory networks from expression profiling and transcription factor binding site identification', *Nucleic Acids Res.*, Vol. 32, pp. 179–188.
77. Dieterich, C., Cusack, B., Wang, H. *et al.* (2002), 'Annotating regulatory DNA based on man–mouse genomic comparison', *Bioinformatics*, Vol. 18, Suppl. 2, pp. S84–90.
78. Bussemaker, H. J., Li, H. and Siggia, E .D. (2001), 'Regulatory element detection using correlation with expression', *Nat. Genet.*, Vol. 27, pp. 167–171.
79. Roven, C. and Bussemaker, H. J. (2003), 'REDUCE: An online tool for inferring *cis*-regulatory elements and transcriptional module activities from microarray data', *Nucleic Acids Res.*, Vol. 31, pp. 3487–3490.
80. Liu, X. S., Brutlag, D. L. and Liu, J. S. (2002), 'An algorithm for finding protein–DNA binding sites with applications to chromatin-immunoprecipitation microarray experiments', *Nat. Biotechnol.*, Vol. 20, pp. 835–839.
81. Conlon, E. M., Liu, X. S., Lieb, J. D. and Liu, J. S. (2003), 'Integrating regulatory motif discovery and genome-wide expression analysis', *Proc. Natl Acad. Sci. USA*, Vol. 100, pp. 3339–3344.