Available online at www.sciencedirect.com
SCIENCE DIRECT®
Neural Networks 19 (2006) 950–962
Neural Networks
www.elsevier.com/locate/neunet

2006 Special Issue

Self-organizing neural networks to support the discovery of DNA-binding motifs

Shaun Mahony[a,b,*], Panayiotis V. Benos[a,c], Terry J. Smith[d], Aaron Golden[d,e]

[a] *Department of Computational Biology, School of Medicine, University of Pittsburgh, Pittsburgh, PA, USA*
[b] *Department of Computer Science, University of Pittsburgh, Pittsburgh, PA, USA*
[c] *Department of Human Genetics, Graduate School of Public Health, and University of Pittsburgh Cancer Institute, School of Medicine, University of Pittsburgh, Pittsburgh, PA, USA*
[d] *National Centre for Biomedical Engineering Science, NUI Galway, Galway, Ireland*
[e] *Department of Information Technology, NUI Galway, Galway, Ireland*

Abstract

Identification of the short DNA sequence motifs that serve as binding targets for transcription factors is an important challenge in bioinformatics. Unsupervised techniques from the statistical learning theory literature have often been applied to motif discovery, but effective solutions for large genomic datasets have yet to be found. We present here three self-organizing neural networks that have applicability to the motif-finding problem. The core system in this study is a previously described SOM-based motif-finder named SOMBRERO. The motif-finder is integrated in this work with a SOM-based method that automatically constructs generalized models for structurally related motifs and initializes SOMBRERO with relevant biological knowledge. A self-organizing tree method that displays the relationships between various motifs is also presented, and it is shown that such a method can act as an effective structural classifier of novel motifs. The performance of the three self-organizing neural networks is evaluated here using various datasets.

Keywords: Self-organizing map; Self-organizing tree; DNA-binding motif identification

1. Introduction

Transcription factors (TFs) are proteins that bind to DNA at *cis*-regulatory sites and regulate gene expression through activating or inhibiting interactions with the transcriptional machinery. Given a collection of DNA regions that are believed to contain common regulatory elements, computational methods aiming to find transcription factor binding sites (TFBSs) typically proceed by identifying short DNA sequence "motifs" that are statistically over-represented in the input. The motif identification problem is notoriously difficult, however, as motifs are short signals (6–20 bp long) that are hidden amongst a great amount of genomic noise (promoter regions are typically thousands of base pairs long). The nature of the TF–DNA interactions is such that TFs can frequently tolerate variation around their "preferred" DNA target sequence. No information is usually available as to the number of individual TFBSs contained in the input sample, nor to the number of different TFs that might have binding sites in the input sample.

Despite the difficulties, numerous motif prediction techniques have become available over the past few years. Many approaches are based on statistical learning theory methods such as expectation-maximization (e.g. MEME (Bailey & Elkan, 1994)) and Gibbs sampling (e.g. AlignACE (Hughes, Estep, Tavazoie, & Church, 2000), Co-Bind (GuhaThakurta & Stormo, 2001) and BioProspector (Liu, Brutlag, & Liu, 2001)). Such methods work through maximum likelihood parameter estimation of the motif model. Neural networks have rarely been applied to the motif-identification problem, one notable exception being ANN-Spec (Workman & Stormo, 2000), where a Perceptron was combined with a Gibbs-sampler to increase

* Corresponding address: 3087 BST3, Department of Computational Biology, University of Pittsburgh, 3501 Fifth Ave., Pittsburgh, PA 15213, USA. Tel.: +1 412 6488688.
E-mail addresses: shaun.mahony@ccbb.pitt.edu (S. Mahony), benos@pitt.edu (P.V. Benos), terry.smith@nuigalway.ie (T.J. Smith), aaron.golden@nuigalway.ie (A. Golden).

0893-6080/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.neunet.2006.05.023

the specificity of the estimated motif models. Alternative motif identification methods have also been proposed, including word enumeration, winnowing, and dictionary construction based methods (Bussemaker, Li, & Siggia, 2000; Gupta & Liu, 2003; Pevzner & Sze, 2000; Rigoutsos & Floratos, 1998; Sinha & Tompa, 2002).

An alternative approach to the motif identification problem can be defined by phrasing it in the terms of a clustering problem. For example, instead of defining the problem in terms of two models (the motif and the background) whose parameters must be estimated by expectation maximization or other such methods, consider the input sequence collection as a set of short overlapping substrings which may be clustered into a number of bins according to sequence similarity. After clustering, each bin would contain an alignment of similar substrings and therefore a motif. Given a large number of bins, a corresponding large number of motifs would be found by the clustering approach. The vast majority of these motifs would not be TFBS motifs, and would instead be due to the background mutation patterns of the genome. Given an appropriate background model, TFBS motifs can be distinguished from motifs that represent background noise.

One unsupervised clustering algorithm suitable for application to the above alternative phrasing of the motif-identification problem is the Self-Organizing Map (SOM) (Kohonen, 1995). We have previously shown that the SOM can be applied to the motif identification problem, and the SOMBRERO (Self-Organizing Map for Biological Regulatory Element Recognition and Ordering) framework resulted (Mahony, Hendrix, Golden, Smith, & Rokhsar, 2005). In our previous publication, it was demonstrated that SOMBRERO's approach to simultaneously characterizing a complete set of motifs for a given dataset helps to separate weak motif signals from large datasets, and improved motif detection performance in real biological datasets was observed.

We aim to demonstrate that self-organizing neural networks can be applied to a wider range of DNA-binding motif related problems than those described by the original SOMBRERO manuscript. A recent trend in motif-discovery aims to encapsulate the properties of motifs from evolutionarily and structurally related transcription factors into generalized motifs named "familial binding profiles" (FBPs; (Sandelin & Wasserman, 2004)). The properties of such FBPs can be employed to constrain motif-finders towards finding particular classes of motifs. FBPs may also be used to classify a novel motif according to structural class.

In this paper, a detailed description of the SOMBRERO algorithm is made, including algorithmic aspects that were neglected in our previous manuscript. In addition, a number of recent algorithmic performance enhancements and parallelization strategies are found to significantly improve the speed of the algorithm without loss of prediction capability. We also show that a SOM can be used to cluster known PSSMs, thereby yielding a set of automatically generated FBPs. It is shown that a SOM clustering of PSSMs can be used as an effective source of prior knowledge for the SOMBRERO motif-finder, and significant performance improvements are observed as a result. These performance enhancements are demonstrated in a eukaryotic genome-scale comparison of motif-finding performance when SOMBRERO and two other popular motif-finders are employed to predict the binding motifs of 77 distinct transcription factors in the yeast genome.

Finally, we present the first application of a self-organizing tree algorithm (SOTA) to the study of FBPs. The SOTA algorithm is used here to visualize the evolutionary relations between TF binding motifs and also to classify novel PSSMs according to TF familial relationships. The three self-organizing neural networks described in this study can interact with each other to produce an effective platform for motif discovery. In summary, self-organizing neural networks are shown to have wide applicability to the motif-identification problem.

2. DNA-binding motif representation

2.1. Position specific scoring matrices

It is usually possible for TFs to bind to a set of related sequences that share some highly conserved positions as well as some more variable positions. Given a collection of binding sites for a particular transcription factor, the general binding preference can be summarized by aligning the sites and converting the alignment to a binding 'motif'. Binding motifs can be represented by various forms. The simplest is the consensus sequence, where the general preference at each position is denoted using a consensus alphabet (e.g. $\Sigma_{\text{consensus}} = \{A, C, G, T, R, Y, M, K, S, W, N\}$, where $R = A$ or G, $Y = C$ or T, $M = A$ or C, $K = G$ or T, $S = C$ or G, $W = A$ or T, and N is any base). Consensus sequences are commonly employed, but the choice of when to use each degenerate letter is somewhat arbitrary (Day & McMorris, 1992). The consensus sequence confers a significant information loss from the original alignment, since it only approximates the frequencies that each base appears in each position of the alignment of binding sites. Consequently, the use of consensus sequences for TFBS prediction in the promoters of new genes is problematic due to the expected high number of false positive predictions.

Another choice for representing binding motifs is the Position Specific Scoring Matrix (PSSM). A PSSM model is a $4 \times \ell$ matrix (ℓ is the length of the DNA motif) in which each column is log-proportional to the number of observations of each nucleotide at this position of the alignment. A sequence logo is a graphical representation of the motif, where the height of the stack of letters at each position in the motif equals the information content at that position, as it is defined in Schneider, Stormo, Gold, and Ehrenfeucht (1986):

$$I_i = 2 + \sum_{b=A}^{T} f_{ib} \log_2 f_{ib}, \tag{1}$$

where f_{ib} is an entry of a normalized PSSM (i.e. the observed relative frequency of the base b at position i of the motif), $b \in \{A, C, G, T\}$ and $i \in \{1, 2, \ldots, \ell\}$. Fig. 1 provides an example of the consensus sequence, the PSSM model, and

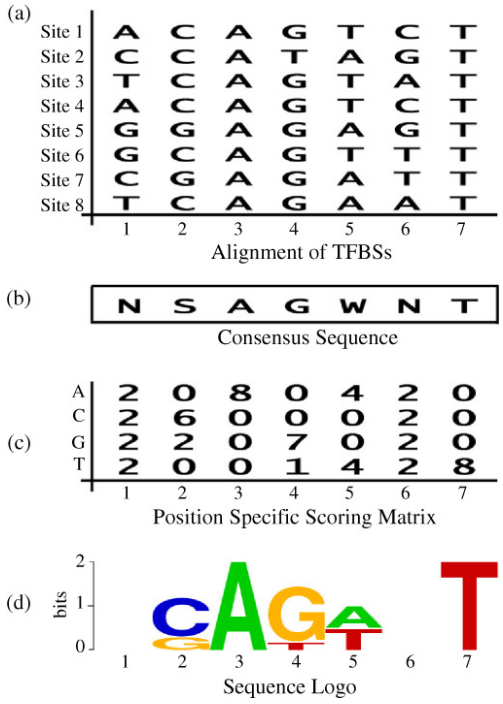

Fig. 1. (a) An alignment of binding sites, (b) a consensus sequence representation of the alignment, (c) a PSSM representation, and (d) a sequence logo representation.

the sequence logo. Similarity between a DNA substring and a PSSM is provided by a log-likelihood ratio score, $S(x)$, defined as

$$S(x) = \sum_{i=1}^{\ell} \sum_{b=A}^{T} x_{ib} \log \frac{f_{ib}}{p_b} \tag{2}$$

where p_b is the background probability for base b and x_{ib}, a position in the indicator matrix for the string x, is 1 if base b is at position i of the string and 0 otherwise. A high score $S(x)$ indicates that the string x is more similar to the motif characterized by the PSSM f than to the background model. It has also been shown that the score $S(x)$ is directly related to the specificity of the interactions between the protein and the DNA (Benos, Bulyk, & Stormo, 2002).

While PSSMs are free from information loss at each position, they assume positional independence between nucleotides, and do not allow for variable spacing between binding nucleotides within a TBFS (insertion/deletion). Both assumptions constitute an approximation of the protein–DNA binding properties. Advanced models that incorporate higher-order interactions between binding positions (e.g. using Bayesian networks, frequency matrices that include pairwise correlations, or variable length Markov models) have proven more effective than the PSSM in representing some protein–DNA interactions (Barash, Elidan, Friedman, & Kaplan, 2003; Osada, Zaslavsky, & Singh, 2004; Zhao, Huang, & Speed, 2005). However, the construction of higher order models requires much larger datasets of known binding sites than do PSSMs, and in any case the improvement in specificity offered by advanced models is often marginal (Benos et al., 2002). Therefore, the most popular binding motif representation is currently the PSSM, and collections of PSSMs constructed using alignments of documented binding sites are stored in databases such as TRANSFAC (Wingender, Dietze, Karas, & Knuppel, 1996) and JASPAR (Sandelin, Alkema, Engstrom, Wasserman, & Lenhard, 2004).

2.2. PSSM comparison and alignment

In Sections 4 and 5, Pietrokovski's methods for aligning two PSSMs are used (Pietrokovski, 1996), but are combined with Sandelin and Wasserman's method for calculating the p-value of an alignment (Sandelin & Wasserman, 2004). Specifically, column-to-column comparisons are made using Pearson's correlation coefficient:

$$r(C, D) = \frac{\sum_{b=A}^{T} (C_b - \overline{C}) \cdot (D_b - \overline{D})}{\sqrt{\sum_{b=A}^{T} (C_b - \overline{C})^2 \cdot \sum_{b=A}^{T} (D_b - \overline{D})^2}} \tag{3}$$

where C_b and D_b are the probability values of base b in columns C and D, respectively, and $\overline{C}$ and $\overline{D}$ are the means of the values in columns C and D, respectively. A modified Smith–Waterman algorithm (Smith & Waterman, 1981) is used to find optimal (gapless) local alignments of PSSM pairs.

In order to compare alignments of different widths, the method for the calculation of empirical p-values described by Sandelin and Wasserman is followed exactly. The method involves extensive analysis with simulated PSSMs to determine the likelihood of any score given the lengths of aligned matrices. The simulated PSSMs reflect the properties of the PSSMs in the JASPAR database (Sandelin et al., 2004). The construction of a dataset of 10,000 simulated matrices follows the instructions on Sandelin and Wasserman's website (http://forkhead2.cgb.ki.se/jaspar/additional/index.htm).

Note that a number of alternative DNA profile comparison methods have been suggested in the literature, including an average log likelihood method (Wang & Stormo, 2003), a position-averaged Kullback–Leibler distance (Aerts, Van Loo, Thijs, Moreau, & De Moor, 2003; Roepcke, Grossmann, Rahmann, & Vingron, 2005), and a method based on the likelihood that aligned columns are independently and identically distributed observations from the same multinomial distribution (Schones, Sumazin, & Zhang, 2005). Sandelin and Wasserman's p-value was chosen for this study as it calculates PSSM similarity in a way that avoids matrix length biases.

3. Finding over-represented motifs using the SOM

3.1. The SOMBRERO motif-finder

SOMBRERO is based on a SOM whose general structure is a two-dimensional (2-D) lattice of interconnected nodes. In SOMBRERO's architecture, PSSMs are embedded as models at each node on the SOM grid. The motif discovery problem aims to find over-represented features of length ℓ in an input dataset of DNA sequences. SOMBRERO, therefore, aims to

align similar ℓ-mer sequences at each SOM node. With this aim in mind, the training algorithm proceeds as follows:

Algorithm 1 (*SOMBRERO*).

1. An $X \times Y$ grid of nodes is created, and the coordinates of the nodes are denoted by $z = (z_1, z_2)$. Each node contains a PSSM model f^z and a count matrix c^z that contains the number of base b observations at each position i in the current alignment.
2. A length ℓ is chosen, typically between 8 and 20, and the input sequences are segmented into every overlapping ℓ-mer $(x_j, j = 1, \ldots, N)$. The PSSM models are initialized using an ordered gradient random initialization, where the PSSMs in each corner of the lattice are biased towards a particular base (and gradients of preference exist in other nodes).
3. Each x_j is assigned to the node with the corresponding maximum likelihood, i.e. the highest score $S_z(x_j)$.
4. Update step:
   4.1. The count matrix c^z is updated for each node, according to the current set of ℓ-mers aligned at the node.
   4.2. New models are generated by augmenting the profile matrix:

$$f^z_{ib} = \frac{\sum_{z'} \Phi(|z - z'|) c^{z'}_{ib} + \beta p_b}{\sum_{b'} \sum_{z'} \Phi(|z - z'|) c^{z'}_{ib} + \beta} \tag{4}$$

   where p_b is the background probability model, β is a small scaling factor that helps to avoid zero probabilities, and $\Phi(|z - z'|)$ is a neighbourhood function that defines the proportion that a node will contribute to another node based on their distance $|z - z'|$ away on the SOM lattice. For our purposes, the Gaussian neighbourhood function is used:

$$\Phi(|z - z'|) = \mathrm{e}^{-[(z_1 - z'_1)^2 + (z_2 - z'_2)^2]/\gamma}. \tag{5}$$

   Here the term γ is a measure of the sharpness of the neighbourhood function and is defined as $\gamma \equiv 1/\log(\delta)$ so that adjacent nodes will contribute $1/\delta$ of their counts to each other. In practice, δ ranges from 4 to 15 over the course of training. Thus, the contributions from f^z_{ib} to the counts of neighbouring nodes initially strongly enforce the similarity of nearby nodes, and end up contributing little at the end of training.
5. Training repeats from step 3 until convergence (defined here as 100 cycles). Once convergence is reached, each string x_j is assigned to its most similar node. In the case where two or more strings at a given node are overlapping strings in the input sequences, only the string with the larger $S_z(x_j)$, is kept. At this point, each node will have a PSSM motif in its final state as well as a list of ℓ-mers that contributed to the motif's construction.
6. Post processing steps:
   6.1. Significant features are distinguished from those that would be expected due to chance. A third-order Markov chain model of the relevant background is used to generate 100 random datasets (each of the same length as the training set), and these sets are used to find the expected number of occurrences and standard deviation of each motif, thus yielding z-scores ($Z_{\text{score}} = (n_{\text{obs}} - \langle n \rangle)/\sigma$) for each node's motif.
   6.2. Repetitive chains of DNA that exist throughout the genome may sometimes be found as over-represented motifs, but are in fact uninteresting from the viewpoint of TF-binding motif identification. Such repetitive motifs are filtered from the output (i.e. not reported to the user even if appearing over-represented) using a motif complexity score, where complexity refers to a measure of the diversity of bases appearing in the PSSM. The complexity score, which is a natural extension of a common single-string score (Wan, Li, Federhen, & Wootton, 2003), is given by

$$C(z) = \left(\frac{1}{4}\right)^{\ell} \prod_{b=A}^{T} \left(\frac{\ell}{\sum_{i=1}^{\ell} f^z_{ib}} \right)^{\sum_{i=1}^{\ell} f^z_{ib}}. \tag{6}$$

   Any nodes which receive less than a reasonably low complexity score (0.01 in this study) are discounted from being treated as a possible functional motif. Note that of the 766 experimentally verified TF binding motifs in version 9.3 of the TRANSFAC database, only 12 (1.56%) were found to have a complexity score less than the threshold employed here. Note also that the above complexity measure may underestimate the complexity of longer motifs, as less deviation from $f_{ib} = 0.25$ is tolerated when ℓ becomes large. However, in the range of typical TF-binding motif lengths the measure was found to serve as a suitable filtering mechanism ($\sim$94% of TRANSFAC motifs are $\leq$20 bp in length).
7. In practice, various motifs of different lengths can exist in a single dataset. Since the methods employed in this algorithm rely on a fixed ℓ for comparisons between ℓ-mers and PSSMs, separate SOMs must be trained from step 2 for various length ℓs. The significance of the motifs found for different values of ℓ are comparable through the z-score of step 6. Since the z-score is length independent, the motif with the highest z-score across all SOMs trained will refer to the most overrepresented motifs from all values of ℓ tested. In this study, separate SOMs are typically trained across all even lengths between 8 and 20.

3.2. Scaling the lattice size for larger datasets

In order to investigate the effect on motif-finding performance of varying the SOM lattice size, various sized SOM lattices were tested on artificial sequence datasets with lengths varying from 1 to 10 kbp. Each sequence dataset was created using a third-order Markov model of yeast (*Saccharomyces cerevisiae*) intergenic sequences, and a random number (mean $\approx$ 15) of *mig1* binding motif instances was placed at random positions in each dataset. The tests

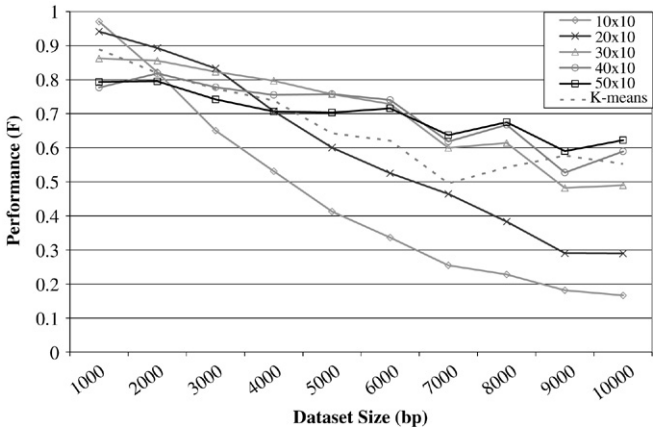

Fig. 2. The effect on SOMBRERO's performance of varying the SOM lattice size as a factor of input dataset size. The performance of a k-means based motif-finder is also shown (where k is set equal to the number of nodes on the best performing SOM in each dataset).

were run twenty times for each dataset size to generate the average performance values displayed in Fig. 2. Performance is measured using the *harmonic mean* (F) of sensitivity (S_N) and specificity (S_P):

$$F = 2(S_N \cdot S_P)/(S_N + S_P). \tag{7}$$

Sensitivity is the percentage of true positive predictions over the number of true sites; *specificity* is the percentage of true positive predictions over the total number of predictions. The value of F ranges from 1, representing perfect recall of the true motif instances with no false positive predictions, to 0, representing no correct prediction found for the motif. A k-means based motif-finder (identical to SOMBRERO in all but the exclusion of neighbourhood contributions in the update step) was also run on the datasets, where k was set equal to the number of neurons on the best performing SOM for each dataset size.

It is expected that the SOM size should be scaled up for larger input data sets. Scaling is necessary because a small SOM trained on a large dataset leads to overcrowding at individual nodes, and thus motif-finding performance (in particular specificity) is heavily reduced. Conversely, a large SOM trained on a small dataset leads to the nodes becoming too specialized, explaining the poorer performance rates in such cases. In this application, the optimum SOM performance is achieved by keeping a ratio in the order of ten input dataset base pairs for every node on the SOM (Fig. 2). Applying this empirically-determined ratio, the following lattice sizes were used in this study: 10×10 nodes for datasets in the interval 0–1999 bp, 20×10 nodes for the interval 2000–3999 bp, 30×15 nodes for the interval 4000–7999 bp, 40×20 nodes for the interval 8000–12,499 bp, and 50×25 nodes for datasets larger than 12,500 bp.

Note that the SOMBRERO architecture uses the SOM as a feature extractor, in contrast to the more common use of large SOMs for data-space visualization. A number of other clustering algorithms may serve the same purpose as the SOM when applied to the motif-finding problem in this context. As detailed in Fig. 2, however, a comparison between the SOM and the k-means algorithm (with the same number of cluster centroids as SOM lattice nodes) in a motif-finding situation shows the SOM to have superior performance. The performance advantages of the SOM may point to the non-equiprobabilistic nature of the final SOM lattice state, as such a property may afford more specificity to nodes containing clusters of highly similar datapoints. In any case, the results show that the neighbourhood function plays an important role in optimizing the motifs found by SOMBRERO.

3.3. SOM optimization and parallelization

One of the major problems of the original SOMBRERO algorithm was its computational cost. The training time was $O(L(MN)+(MN)^2)$ when an $M \times N$ SOM was applied to a data set of total length L. Originally, the Gaussian contribution from every other node on the lattice was calculated for every node that was undergoing an update, resulting in the $(MN)^2$ term in the time-cost equation. A simple calculation shows that the original neighbourhood update is computationally wasteful. For the Gaussian formula described in Eq. (5), and with δ varying from 4 to 15, it can be shown that only those nodes within a radius of five nodes have greater than 10^{-10} contribution to the node being updated at the start of training. Making the obvious optimization of only calculating neighbourhood contributions from those nodes within a radius of five nodes significantly speeds the training process, and has not been observed to affect the motif-finding accuracy of SOMBRERO in any way.

The second optimizing adjustment made to SOMBRERO's algorithm uses a modified winning node search routine. An exhaustive winning node search routine (i.e. searching through all nodes on the lattice) is used for the first 20% of training cycles. Thereafter, the search procedure searches only against the previous winning node and its immediate neighbours. If the local winning node is not identical to that found in the previous cycle, the search is performed on the current local winner and its neighbours. The exhaustive search procedure is used every tenth training cycle in order to smooth out any local maxima. Again, no adverse effects on motif-finding accuracy were observed as a result of the above adjustment, while the resulting speedup of training time was considerable. Fig. 3 demonstrates the effect on the SOMBRERO training phase time-cost (on a single processor) resulting from the two optimizations described above. It should be noted that both optimizations follow suggestions originally described by Kohonen (1995).

In order to reduce the apparent time-cost for those users who have access to multi-processor computing facilities, training set parallelism was implemented on the SOMBRERO C++ code using the Message Passing Interface (MPI). Training set parallelism allows each processing unit to contain separate copies of the SOM (synchronized periodically) and the training set is subdivided across each. The batch-learning SOM algorithm is intuitively amenable to training set parallelization. In SOMBRERO's MPI-enabled implementation, the training phase (Algorithm 1, steps 3–5) and the phase which maps random DNA to the trained SOM (Algorithm 1, step 6.1)

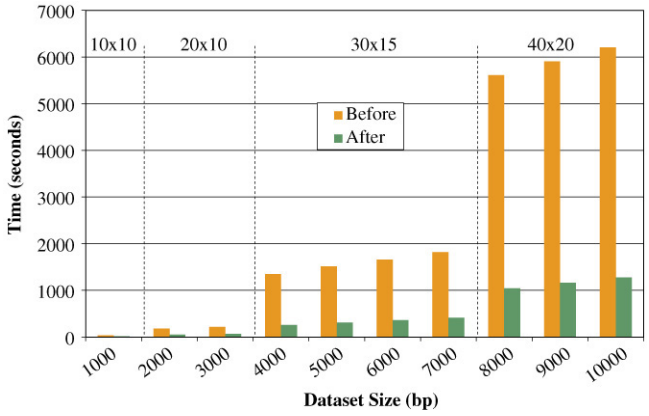

Fig. 3. The time taken to complete SOMBRERO's training phase for various sized datasets before and after implementation of the optimizations. The points at which different sized SOM lattices are used are labelled with dashed lines.

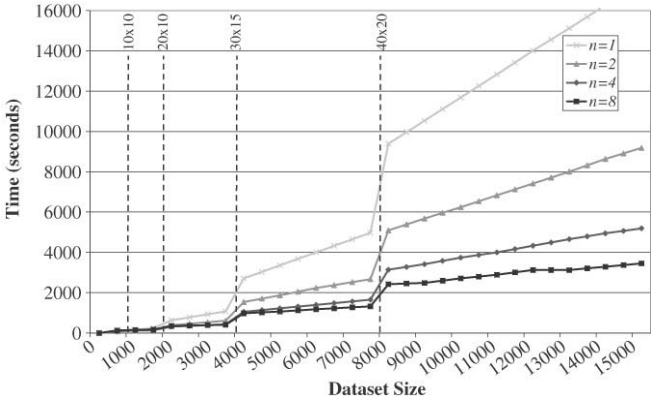

Fig. 4. Timing information for SOMBRERO running with various numbers of processors n on different dataset sizes. The points at which different SOM lattice sizes are used are shown with dotted lines.

are parallelized separately. The latter is the most successfully parallelized, as this phase requires a minimum of inter-processor communication until all sequences are mapped. The resulting speedup for mapping random DNA to the SOM is close to $1/n$, where n is the number of processors used. In parallelizing SOMBRERO's training phase, however, the SOMs at each processor must be consolidated after each training iteration. The ensuing inter-processor communications overhead does not allow optimally efficient speedup through the use of more processors. The neighbourhood update step (Eq. (4)) is also parallelized; for each node being updated, each processor calculates the new models for a subset of SOM nodes, and the SOM is again consolidated before the new training iteration begins. Fig. 4 demonstrates the overall speedup resulting from parallelization as a factor of the number of processors used for various dataset sizes, where the MPI-enabled version of SOMBRERO is running on an SGI Origin 3800 supercomputer.

4. Clustering motifs using the SOM: Automatic generation of familial binding profiles

4.1. Familial binding profiles as priors for motif-finders

In the original description of the SOMBRERO algorithm, randomized and ordered SOM lattice initialization strategies were explored (Mahony, Hendrix et al., 2005). Although no significant difference in motif-finding accuracy was observed between the two strategies, the ordered initialization was chosen for the smoothness it introduced into the SOM training procedure and its theoretical stability.

An alternative approach to SOMBRERO initialization has since been developed (Mahony, Golden, Smith, & Benos, 2005). Many known binding motifs exist in databases such as TRANSFAC (Wingender et al., 1996), and the binding preferences displayed in these motifs are not randomly distributed. For example, if two transcription factors are related evolutionarily, then frequently their corresponding DNA-binding motifs also display some similarity. For particular families of related transcription factors, so-called familial binding profiles (FBPs) can be defined, and they represent the average or generalized binding preference within that family. One of the current challenges in motif-finding application development is to incorporate the knowledge provided by such FBPs into methods that aim to find novel motifs in a set of sequences. However, even the construction of FBPs is problematic, with previous studies taking a manual approach to clustering related PSSMs into generalized models (Sandelin & Wasserman, 2004).

It has previously been demonstrated that incorporating a motif as a "biasing prior" for a motif-finder based on Expectation Maximization or Gibbs-sampling improves the detection of other motifs related to the prior (Bailey & Elkan, 1995; Sandelin & Wasserman, 2004). Foreknowledge of the familial membership of an unknown motif is rarely available, however, and incorporating an incorrect prior has a detrimental effect on motif-finding performance. Current motif-finding methods can only incorporate a single prior in a given motif-finding run, and therefore the choice of biasing prior is critical. However, the SOMBRERO lattice contains many PSSM models, and thus the opportunity exists for multiple priors to be used to initialize the lattice. One effective way to order a set of known PSSMs into a structure suitable for initializing and biasing SOMBRERO is to train another SOM (with identical lattice dimensions) on the set of PSSMs and use the final node states from that SOM as the initial SOMBRERO node states.

4.2. The binding profile SOM

The following SOM training algorithm is used to automatically organize a set of PSSM models into a number of FBPs. Subsequently, the FBPs can be used as initial states for the standard SOMBRERO algorithm. The main conceptual difference between the following algorithm and Algorithm 1 is that here PSSM models and not ℓ-mers are to be clustered at the nodes. The training algorithm for the binding profile SOM (BP-SOM) proceeds as follows:

Algorithm 2 (*The Binding Profile SOM*).

1. The BP-SOM lattice size is set equal to the size required by the SOMBRERO grid, and each node model m_j is initialized as a PSSM with random values.

2. For each training set PSSM, x_i ($i = 1, \ldots, N$):
   2.1. x_i is aligned to every SOM node model m_j using the alignment method described in Section 2.2.
   2.2. The node w whose model m_w has the best p-value alignment score to x_i is selected, and the PSSM x_i is assigned to that node.
3. Update step:
   3.1. At each node j, all clustered PSSMs are aligned against each other. Given the average p-value (p_v) obtained in comparisons of profile v to all other PSSMs at the same node, the weight (Z_v) of each PSSM is calculated as $Z_v = 1 - p_v$. The PSSM with the highest Z_v is designated as the alignment positioning template.
   3.2. At each node, a new binding profile is generated according to the equation

$$m_j(t+1) = \sum_{i=1}^{N} align(x_{i,k} \cdot Z_{i,k} \cdot e^{-|j-k|^2/\gamma}) \qquad (8)$$

   where *align*() is a function that aligns the columns of each x_i (clustered at node k) to the relevant alignment positioning template at node j, and $|j - k|$ is the distance on the SOM grid between nodes j and k. The Gaussian sharpness factor, γ, is defined as before, but here δ ranges from 4 to 30 during training. The length of the new model depends on the quality of the alignment. Flanking columns with low information content (<0.4 bits) are excluded from the new model, to a minimum model length of 8 columns.
4. The training process repeats from step 2 until convergence (to a maximum of 100 cycles).

4.3. Combining the BP-SOM and SOMBRERO: Improved motif-finding performance in artificial datasets

The BP-SOM algorithm results in an ordered grid of FBP models that can be used as the initial states for a SOMBRERO grid of equal size. The FBPs generated by the BP-SOM will be of various lengths, so shortening or padding is applied as appropriate in order to make the BP-SOM PSSMs compatible with SOMBRERO's length ℓ models. The effect of incorporating prior knowledge into SOMBRERO's initialization is demonstrated here using various artificial datasets. The artificial datasets were constructed using a third-order Markov model of yeast intergenic sequence. The datasets contained various instances of one of four different motifs: the mammalian motifs CREB and E4BP4, and the yeast motifs CSRE and GAL4 (each PSSM was procured from TRANSFAC). For each motif, 200 datasets were constructed. Each dataset contained sequences to a total sequence length of between 1 and 10 kbp per dataset. A random number of the relevant motif instances (the mean occurrence for each motif was ∼15) was placed at random intervals in each dataset.

Three SOMBRERO initialization strategies are tested on the artificial datasets: the original gradient-random initialization, a completely random initialization, and an initialization based on BP-SOM priors. For the cases where SOMBRERO uses a prior, the prior refers to the end state of a BP-SOM that has been previously trained on a selection of 257 mammalian-specific PSSMs (taken from TRANSFAC and JASPAR).

Fig. 5 presents the results of the analysis, where the percentage performance difference of each initialization strategy is compared (with the gradient-random initialization's performance serving as the baseline). As in Section 3.2, the performance is again defined as the harmonic mean of sensitivity and specificity. There is little variation between the average performance of the gradient random initialization and the random initialization in any of the four datasets, and this is in line with previous observations (Mahony, Hendrix et al., 2005). However, the use of the prior initialization improves motif-finding performance over the other initialization strategies for the two motifs that are present in the prior set (CREB & E4BP4), and the improvement lasts consistently as the dataset size increases. As expected, no improved performance is observed through the use of a mammalian-specific prior for either the GAL4 or CSRE motif datasets, as no related motifs are included in the prior dataset. However, neither does the use of the mammalian-specific prior initialization adversely affect performance when finding these yeast motifs.

In conclusion, this test shows that the use of FBPs as priors in the SOMBRERO algorithm can improve its performance when the searched motifs are included in the FBPs, while it does not decrease its performance when they are not.

4.4. Improved motif-finding performance in S. cerevisiae regulatory regions

In order to determine SOMBRERO's motif-finding performance in real genomic datasets, a large-scale assessment was carried out using datasets taken from the *S. cerevisiae* genome. Harbison et al. have determined the genomic occupancy of 203 transcription factors (i.e. the intergenic regions bound by each TF) under a variety of environmental conditions (Harbison et al., 2004). Taking only high-confidence ($P \leq 0.001$) location information, and restricting interest to those binding sites that are evolutionarily conserved in at least two other yeast species, Harbison et al. defined precise binding locations for 102 transcription factors.

The Harbison et al. data (available from http://jura.wi.mit.edu/fraenkel/download) was converted into a form suitable for assessing the accuracy of motif-finders. For each of the 77 transcription factors that have more than four high-confidence binding locations recorded, the entire intergenic sequence that encapsulates each recorded binding site was taken from the yeast genome. Therefore, 77 datasets to a total sequence length of 1.67 Mbp were constructed. MEME, AlignACE and SOMBRERO (using both the gradient-random initialization and a prior initialization based on a BP-SOM trained using Harbison et al.'s set of 102 yeast motifs) were each run on the datasets. SOMBRERO and AlignACE were run using default settings. MEME was allowed to search both strands for up to 20 motifs, each of which can occur zero or more times in each sequence. AlignACE also requires an expected

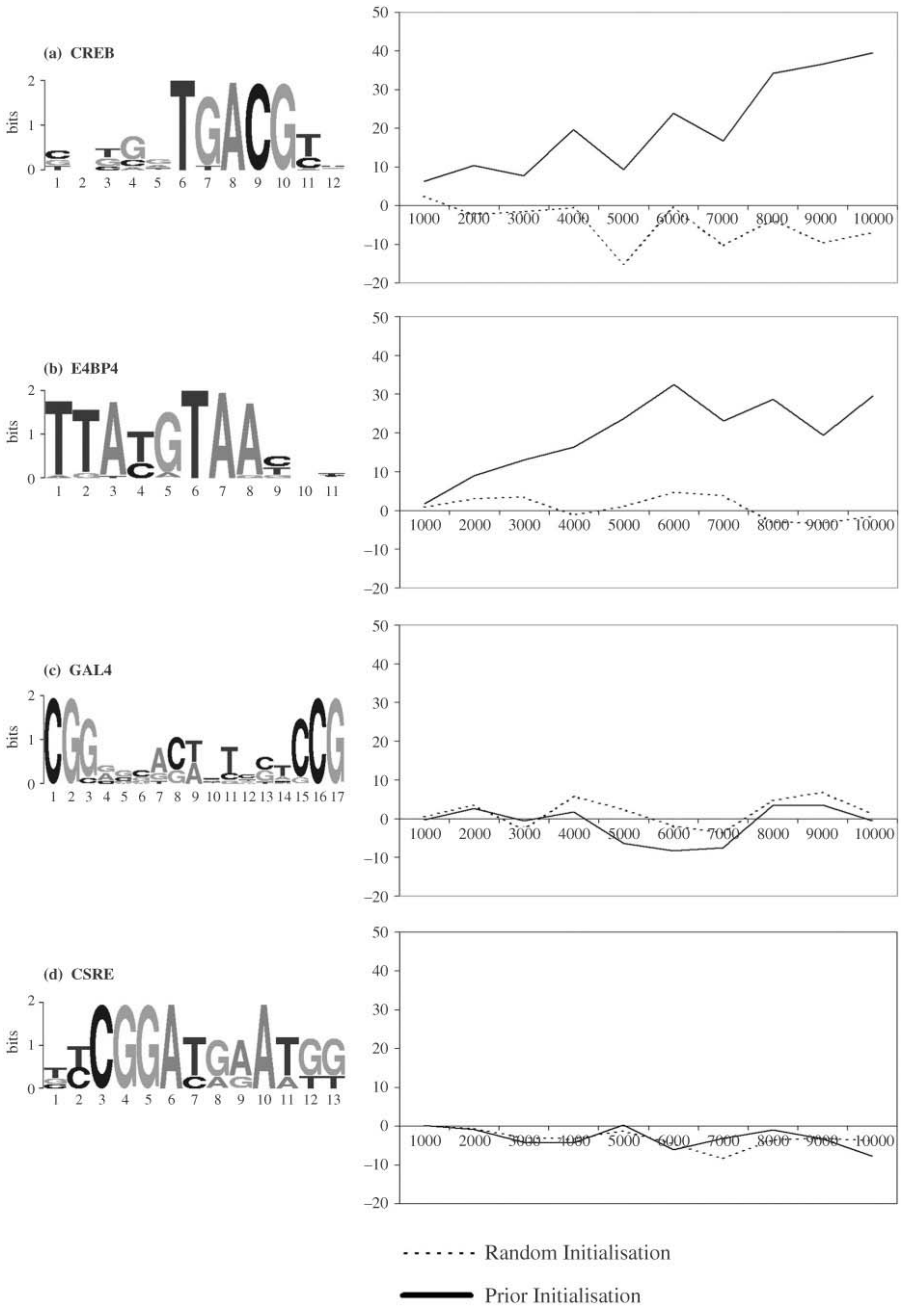

Fig. 5. The average percentage difference in performance of the random initialization and the prior initialization in comparison with the gradient-random initialization's performance (baseline) in various artificial datasets. The graphs plot the percentage change in average harmonic mean against dataset size (in bp).

motif length, and the correct motif length was provided for each case. SOMBRERO's lattice size was chosen according to the heuristic guidelines given in Section 3.2. In every case, accuracy of motif-finding is judged in relation to the best matching motif found in the top 20 results from each method.

Every effort has been made to ensure that the comparison between the various motif-finding programs is fair. However, the conclusions from such comparisons should always be interpreted with caution since the very nature of the algorithms and the input parameters make such comparisons somewhat "biased". Furthermore, it should be noted that neither MEME nor AlignACE allow priors to be used in the sense that SOMBRERO allows. Therefore, the results of the comparisons should serve only as a frame of reference. Finally, the high stringency of Harbison et al.'s methodology means that while each recorded binding site is highly likely to be bound *in vivo*, there may also be an unknown number of other functional binding sites present in each dataset. This should be particularly noted when comparing each method's false positive rates.

The results of the analysis are summarized in Table 1, and are described in terms of average false negative (FN), false positive (FP) and harmonic mean (F) across various intervals of dataset size. In Table 1, the 77 datasets are

Table 1
Summary of the performance of motif-finders in 77 yeast datasets

Dataset size	Sets	Ttl. size (bp)	Sites	SOMBRERO (G.-R. Init.)			SOMBRERO (Prior Init.)			MEME			AlignACE		
				Avg. FN	Avg. FP	Avg. F	Avg. FN	Avg. FP	Avg. F	Avg. FN	Avg. FP	Avg. F	Avg. FN	Avg. FP	Avg. F
40–85 kbp	16	885 019	1751	0.26	0.34	0.69	0.22	0.31	0.72	0.59	0.18	0.46	0.44	0.36	0.54
20–40 kbp	14	487 135	940	0.38	0.40	0.60	0.22	0.35	0.69	0.56	0.38	0.48	0.42	0.39	0.51
10–20 kbp	16	224 200	408	0.43	0.48	0.53	0.25	0.37	0.68	0.51	0.37	0.54	0.41	0.55	0.45
5–10 kbp	15	117 712	209	0.37	0.61	0.47	0.20	0.59	0.53	0.39	0.26	0.63	0.52	0.50	0.55
0–5 kbp	16	54 723	114	0.24	0.63	0.47	0.20	0.69	0.41	0.44	0.44	0.49	0.64	0.61	0.47
Total	**77**	**1676 760**	**3289**	**0.33**	**0.43**	**0.62**	**0.24**	**0.41**	**0.67**	**0.60**	**0.29**	**0.51**	**0.52**	**0.31**	**0.57**

arbitrarily grouped according to size. Across the entire 77 datasets, SOMBRERO incorporating a yeast-specific prior has the best performance rate and lowest false negative rate of any of the motif-finders, and the use of a prior significantly improves upon the overall accuracy of the original SOMBRERO initialization strategy. Complete performance results for each of the 77 individual datasets is available in Supplementary Table 1 (http://biodev.hgen.pitt.edu/services.html). SOMBRERO incorporating a yeast-specific prior has the best performance rate in 37 of the 77 individual datasets, while SOMBRERO without a prior performs best in 10, MEME in 15 and AlignACE in 15 datasets.

5. PSSM classification using self-organizing trees

The BP-SOM described in Section 4.2 is clearly unsuitable for visualizing familial relationships between a set of PSSMs. The output lattice does not suggest the form of the relationship between any two nodes, the number of clusters (or FBPs) found by the SOM is dependant on the lattice size, and empty nodes representing no PSSM family often remain on the lattice at the end of training. The BP-SOM is thus useful only in the context of organizing known DNA-binding motifs into a data structure that can initialize SOMBRERO. However, there may be a biological interest in determining relationships between motifs and motif families on the basis of sequence similarity and divergence. Such a study would ideally make use of a tree formalism (as commonly implemented in phylogeny studies) as opposed to the 2-D lattice offered by the SOM. In this section, the Self-Organizing Tree Algorithm (SOTA (Dopazo & Carazo, 1997)) is used to assess the applicability of self-organizing trees to the study of familial relationships between motifs.

The SOTA was first described as a growing cell structure approach to automatically constructing a phylogenetic tree for a set of protein sequences (Dopazo & Carazo, 1997). Here the SOTA is applied to the hierarchical clustering of PSSMs, and therefore the SOTA nodes and cells each contain a PSSM that evolves over the training period to represent a PSSM or set of PSSMs from the input dataset. The topology of the SOTA neural network takes the form of a binary tree. The tree begins with two external elements, denoted as cells, connected by an ancestor, named a node. Training proceeds similarly to the SOM algorithm, but at the end of a training cycle the tree grows by splitting one cell. The tree stops growing when a predefined threshold has been reached, or when every cell has a single datapoint clustered within (as in our usage). The novel algorithm used to cluster PSSMs using SOTA in this study is summarized below and will henceforth be referred to as the "binding profile SOTA" (BP-SOTA).

Algorithm 3. The binding profile SOTA

1. Two cells, and a connecting node, are initialized as random value PSSMs m_j.
2. For each input PSSM, x_i $(i = 1, \ldots, N)$:
   2.1. x_i is aligned to every cell on the tree (m_j) using the alignment method described in Section 2.2.
   2.2. The winning cell is chosen as the cell w whose model m_w has the best p-value score to x_i.
   2.3. x_i is clustered at cell w.
3. The update step only applies to cells, their immediate ancestor node (the "mother") and the other cell descended from the same mother (the "sister"). New models are generated according to

$$m_j(t+1) = m_j(t) + \sum_i^N align(x_{i,k} \cdot Z_{i,k} \cdot \eta_j), \tag{9}$$

   where the notation follows that of the BP-SOM, and the learning rate $\eta_j = \alpha_i(1 - t/M_t)$, where $\alpha_{\text{sister}} = 1/2$, $\alpha_{\text{mother}} = 1/8$ and $\alpha_{\text{other}} = 0$.
4. The training process repeats from step 2 until M_t cycles are reached ($M_t = 50$).
5. Growing phase. If the algorithm has not yet converged, the cell with the lowest weight (Z_j, defined in Algorithm 2) is split, giving rise to two (initially) identical descendants.
6. Training repeats from step 2 until convergence. Convergence is defined here as the point where every cell contains one and only one PSSM, although training can be stopped at any point during the growth of the tree.

The application of the SOTA to the study of PSSMs allows the full set of similarity relationships between various PSSMs to be represented on a binary tree structure. As originally noted by Sandelin and Wasserman, familial binding profiles can be used to help predict the protein structural class for a novel motif (Sandelin & Wasserman, 2004). The tree created by the BP-SOTA is also suitable for classifying newly discovered motifs. The classification power of the tree was measured using a leave-one-out cross-validation study; each of the PSSMs in a JASPAR subset (which includes only those PSSMs belonging to families represented by five or more members in the JASPAR database:

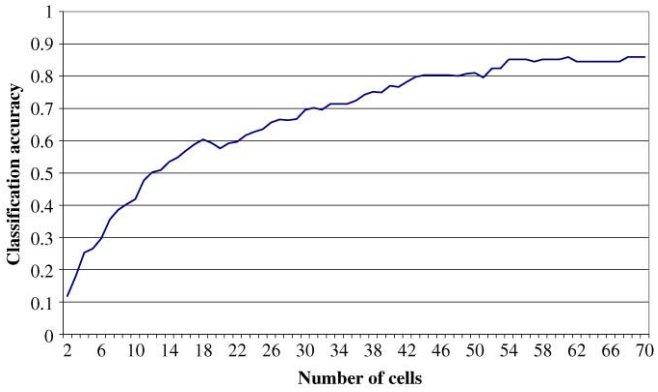

Fig. 6. Classification accuracy of BP-SOTA as a function of the number of cells on the tree.

71 PSSMs belonging to 11 families) is removed in turn, and the BP-SOTA is run on the remaining 70 profiles. The left-out PSSM is compared to all cells on a given level of the tree, and classification power is measured by calculating the percentage of members of the most similar cell that are of the same structural family as the test PSSM. Fig. 6 shows the average classification power measured for each level of growth in the tree (the 70 cell level represents completion of the tree construction; each cell represents a single PSSM).

As can be seen from the figure, the classification power of the BP-SOTA is 86%, which is equivalent to the prediction power of Sandelin and Wasserman's manually constructed FBPs in the same dataset (87%). Interestingly, the tree does not have to be fully formed for this accuracy to be obtained. The accuracy limit is reached when the BP-SOTA tree consists of only 54 cells, which suggests an underlying familial structure has been found at this point. Perfect accuracy of PSSM classification may be unobtainable, as the binding preferences of some members of a TF family might not be similar to other members of the same family.

6. The integrated SOMBRERO motif-finding platform

A complete motif-finding platform can be defined by combining the three self-organizing neural networks described above. As described in Section 4.2, the BP-SOM can be used to provide a source of prior knowledge for the SOMBRERO motif-finder. As a consequence of SOMBRERO repeating the motif search over various values of ℓ, slightly different instances of the same motif may be discovered and reported as distinct motifs. The third subsystem, the BP-SOTA, can be employed in order to point out similarities between the discovered motifs to the user.

The functionality of the complete three-level SOMBRERO motif-finding system is briefly demonstrated here. In this demonstration, SOMBRERO aims to identify motifs in an artificial dataset, generated using a third-order Markov model of yeast intergenic DNA. Within the artificial dataset are implanted 10 TFBSs for each of three TFs: GAL4, NF-κB and CREB. SOMBRERO's grid is initialized using a BP-SOM that has been previously trained on a collection of 257 mammalian PSSMs taken from the TRANSFAC and JASPAR databases. The collection includes the NF-κB and CREB PSSMs, but not the GAL4 motif (nor is any motif similar to GAL4 present in the mammalian dataset). A portion of the trained BP-SOM is displayed in Fig. 7(a). Node 2, 7 contains a FBP that represents seven members of the REL family of TFs, including the NF-κB motif.

The final state of the BP-SOM is used to initialize a 20×10 SOMBRERO grid. The input sequence dataset of 2000 bp is divided into ℓ-mers and clustered on the grid. Training repeats for all even values of ℓ between 8 and 18. The grid portion in Fig. 7(b) shows some final nodes states on a SOMBRERO grid of $\ell = 12$. Note that the motif in node 2, 7 has changed very little from the initial state. The NF-κB motif is present in the input dataset, and thus it reinforces the presence of the motif at node 2, 7 on the SOMBRERO grid. Contrast this with node 1, 8, whose motif has changed drastically from the initial state, due to the non-presence of the relevant motif in the input sequences.

The significance of every motif existing in the final SOMBRERO grids is calculated. Occurrences of the same motif may have been found in separate SOMBRERO grids that used different values of ℓ. In order to illustrate the relationships between various motifs for the user, the top 15 scoring motifs are clustered using the BP-SOTA. The resulting tree is shown in Fig. 7(c). The BP-SOTA properly separates the motifs on the basis of the represented transcription factor.

7. Conclusion

Self-organizing neural networks have been previously applied to the study of DNA sequence data. For example, the SOM and related algorithms have been applied in the context of codon usage and genome signature analysis (Abe, Kanaya, & Kinouchi, 2002, 2003; Hayashi, Abe, & Sakamoto, 2005; Kanaya, Kinouchi, & Abe, 2001; Wang, Badger, & Kearney, 2001), gene prediction (Gorban, Zinovyev, & Popova, 2003; Gorban, Zinovyev, & Wunsch, 2003; Mahony, McInerney, Smith, & Golden, 2004), endogenous retrovirus clustering (Oja, Sperber, Blomberg, & Kaski, 2004, 2005), various classification problem (Aires-de-Sousa & Aires-de-Sousa, 2003; Naenna, Bress, & Embrechts, 2003; Wang, Azuaje, & Black, 2004), and even the efficient design of DNA microarrays for SNP analysis (Douzono, Hara, & Noguchi, 2001).

In the domain of motif-finding, Arrigo et al. attempted to use a SOM to find "singular" or unusual patterns in promoter sequences by using the Tanimoto measure to find the DNA sequence most distant from the final SOM weight vectors (Arrigo, Giuliano, Scalia, Rapallo, & Damiani, 1991). The patterns thus discovered were usually GC-rich, but this had probably more to do with the data representation method (based on the conversion of bases to ordinal numbers) than with the regulatory potential of the sequences. By incorporating PSSMs as node models, the SOMBRERO motif-finder described in Section 3 offers a much more robust approach to finding regulatory motifs, and allows a SOM-based feature detector to outperform methods based on statistical learning theory.

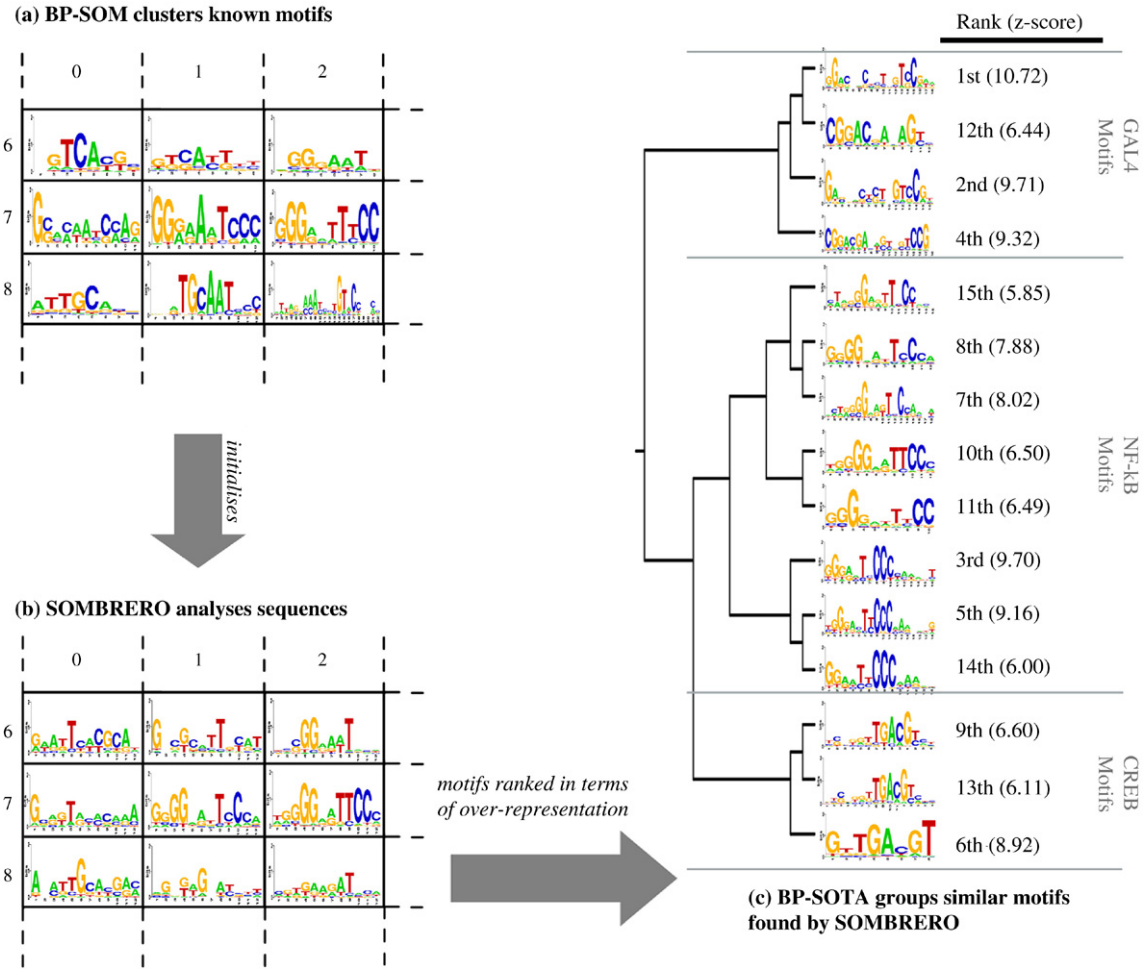

Fig. 7. Illustration of the cooperation between three self-organizing neural networks in the complete motif-finding platform.

Section 4 described a SOM-based approach to automatically constructing familial binding profiles. Cartharius et al. also used the SOM to cluster known PSSMs into families for use with their MatInspector TFBS prediction program (Cartharius et al., 2005). In their case, however, the PSSM were first converted into vectors of di-, tri- and tetra-nucleotide frequency within the matrix. The short length of most PSSMs means that the frequency vector representation will suitably capture the features of the matrices, but alignment methods that do not require vectorization of the PSSMs are preferable. Through the use of accurate PSSM alignment methods, therefore, the BP-SOM is expected to automatically construct FBPs more accurately than the approach of Cartharius et al.

The BP-SOM was shown to be an effective means of incorporating prior biological knowledge into the SOMBRERO motif-finder. The provided examples demonstrate that the use of prior biological knowledge with SOMBRERO gives the type of improved performance that is desired; motif-finding performance is improved if the relevant motif is present in the prior, and performance is not negatively affected for those motifs or structural classes that are not represented in the prior. SOMBRERO is currently the only motif-finder that facilitates the incorporation of a large set of prior biological knowledge.

Finally, a self-organizing tree algorithm was shown to be an effective classifier of structural class for novel binding motifs. It will be interesting in the future to compare the results obtained with SOTA with the methods that have been used in the past in the study of protein evolution, such as UPGMA (Michener & Sokal, 1957) and neighbour-joining (Saitou & Nei, 1987), as well as probabilistic approaches such as maximum parsimony (Fitch, 1971) and maximum likelihood (Felsenstein, 1973). The lack of large-scale DNA-binding motif datasets means that the study of familial binding profiles is in its infancy. However, as shown in this work, self-organizing neural networks have wide applicability to the study of DNA-binding motifs and familial binding profiles, and will play an important role in their future study.

Acknowledgements

The authors wish to thank the SFI/HEA Irish Centre for High-End Computing and the NUI, Galway Centre for High-End Computing for their generous provision of computational facilities. The comments of two anonymous reviewers were highly appreciated. PVB was supported by NSF (grant number MCB0316255).

References

Abe, T., Kanaya, S., Kinouchi, M., Ichiba, Y., Kozuki, T., & Ikemura, T. (2002). A novel bioinformatic strategy for unveiling hidden genome signatures of eukaryotes: Self-organizing map of oligonucleotide frequency. *Genome Informatics*, *13*, 12–20.

Abe, T., Kanaya, S., Kinouchi, M., Ichiba, Y., Kozuki, T., & Ikemura, T. (2003). Informatics for unveiling hidden genome signatures. *Genome Research*, *13*(4), 693–702.

Aerts, S., Van Loo, P., Thijs, G., Moreau, Y., & De Moor, B. (2003). Computational detection of cis-regulatory modules. *Bioinformatics*, *19*(Suppl 2), II5–II14.

Aires-de-Sousa, J., & Aires-de-Sousa, L. (2003). Representation of DNA sequences with virtual potentials and their processing by (SEQREP) Kohonen self-organizing maps. *Bioinformatics*, *19*(1), 30–36.

Arrigo, P., Giuliano, F., Scalia, F., Rapallo, A., & Damiani, G. (1991). Identification of a new motif on nucleic acid sequence data using Kohonen's self-organizing map. *Computer Applications in Biosciences*, *7*(3), 353–357.

Bailey, T. L., & Elkan, C. (1994). Fitting a mixture model by expectation maximization to discover motifs in biopolymers. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology*, *2*, 28–36.

Bailey, T. L., & Elkan, C. (1995). The value of prior knowledge in discovering motifs with MEME. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology*, *3*, 21–29.

Barash, Y., Elidan, G., Friedman, N., & Kaplan, T. (2003). Modeling dependencies in protein-DNA binding sites. In *Proc. seventh annual inter. conf. on computational molecular biology (RECOMB)* (pp. 28–37).

Benos, P. V., Bulyk, M. L., & Stormo, G. D. (2002). Additivity in protein-DNA interactions: How good an approximation is it? *Nucleic Acids Research*, *30*(20), 4442–4451.

Bussemaker, H. J., Li, H., & Siggia, E. D. (2000). Building a dictionary for genomes: identification of presumptive regulatory sites by statistical analysis. *Proceedings of the National Academy of Sciences of the United States of America*, *97*(18), 10096–10100.

Cartharius, K., Frech, K., Grote, K., Klocke, B., Haltmeier, M., Klingenhoff, A., et al. (2005). MatInspector and beyond: Promoter analysis based on transcription factor binding sites. *Bioinformatics*, *21*(13), 2933–2942.

Day, W. H., & McMorris, F. R. (1992). Critical comparison of consensus methods for molecular sequences. *Nucleic Acids Research*, *20*(5), 1093–1099.

Dopazo, J., & Carazo, J. M. (1997). Phylogenetic reconstruction using an unsupervised growing neural network that adopts the topology of a phylogenetic tree. *Journal of Molecular Evolution*, *44*(2), 226–233.

Douzono, H., Hara, S., & Noguchi, Y. (2001). A design method of DNA chips for SNP analysis using self-organizing maps. In *Proceedings of the international joint conference on neural networks, 2001*.

Felsenstein, J. (1973). Maximum-likelihood estimation of evolutionary trees from continuous characters. *American Journal of Human Genetics*, *25*(5), 471–492.

Fitch, W. M. (1971). Toward defining the course of evolution: Minimum change for a specified tree topology. *Systematic Zoology*, *20*, 406–416.

Gorban, A. N., Zinovyev, A. Y., & Popova, T. G. (2003). Seven clusters in genomic triplet distributions. *In Silico Biology*, *3*(4), 471–482.

Gorban, A. N., Zinovyev, A. Y., & Wunsch, D. C. (2003). Application of the method of elastic maps in analysis of genetic texts. In *Proceedings of the international joint conference on neural networks, 2003*.

GuhaThakurta, D., & Stormo, G. D. (2001). Identifying target sites for cooperatively binding factors. *Bioinformatics*, *17*(7), 608–621.

Gupta, M., & Liu, J. S. (2003). Discovery of conserved sequence patterns using a stochastic dictionary model. *Journal of the American Statistical Association*, *98*(461), 55–66.

Harbison, C. T., Gordon, D. B., Lee, T. I., Rinaldi, N. J., Macisaac, K. D., Danford, T. W., et al. (2004). Transcriptional regulatory code of a eukaryotic genome. *Nature*, *431*(7004), 99–104.

Hayashi, H., Abe, T., Sakamoto, M., Ohara, H., Ikemura, T., Sakka, K., et al. (2005). Direct cloning of genes encoding novel xylanases from the human gut. *Canadian Journal of Microbiology*, *51*(3), 251–259.

Hughes, J. D., Estep, P. W., Tavazoie, S., & Church, G. M. (2000). Computational identification of cis-regulatory elements associated with groups of functionally related genes in *Saccharomyces cerevisiae*. *Journal of Molecular Biology*, *296*(5), 1205–1214.

Kanaya, S., Kinouchi, M., Abe, T., Kudo, Y., Yamada, Y., Nishi, T., et al. (2001). Analysis of codon usage diversity of bacterial genes with a self-organizing map (SOM): Characterization of horizontally transferred genes with emphasis on the *E. coli* O157 genome. *Gene*, *276*(1–2), 89–99.

Kohonen, T. (1995). *Self-organizing maps*. Berlin: Springer-Verlag.

Liu, X., Brutlag, D. L., & Liu, J. S. (2001). BioProspector: Discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. *Pacific Symposium on Biocomputing*, 127–138.

Mahony, S., Golden, A., Smith, T. J., & Benos, P. V. (2005). Improved detection of DNA motifs using a self-organized clustering of familial binding motifs. *Bioinformatics*, *21*, i283–i291.

Mahony, S., Hendrix, D., Golden, A., Smith, T. J., & Rokhsar, D. S. (2005). Transcription factor binding site identification using the self-organizing map. *Bioinformatics*, *21*(9), 1807–1814.

Mahony, S., McInerney, J. O., Smith, T. J., & Golden, A. (2004). Gene prediction using the Self-Organizing Map: Automatic generation of multiple gene models. *BMC Bioinformatics*, *5*(1), 23.

Michener, C. D., & Sokal, R. R. (1957). A quantitive approach to a problem in classification. *Evolution*, *11*, 130–162.

Naenna, T., Bress, R. A., & Embrechts, M. J. (2003). DNA classifications with self-organizing maps (SOMs). In *Proceedings of the 2003 IEEE international workshop on soft computing in industrial applications, 2003*.

Oja, M., Sperber, G., Blomberg, J., & Kaski, S. (2004). Grouping and visualizing human endogenous retroviruses by bootstrapping median self-organizing maps. In *Proceedings of the 2004 IEEE symposium on computational intelligence in bioinformatics and computational biology, 2004*.

Oja, M., Sperber, G. O., Blomberg, J., & Kaski, S. (2005). Self-organizing map-based discovery and visualization of human endogenous retroviral sequence groups. *International Journal of Neural Systems*, *15*(3), 163–179.

Osada, R., Zaslavsky, E., & Singh, M. (2004). Comparative analysis of methods for representing and searching for transcription factor binding sites. *Bioinformatics*, *20*(18), 3516–3525.

Pevzner, P. A., & Sze, S. H. (2000). Combinatorial approaches to finding subtle signals in DNA sequences. *Proceedings of the International Conference on Intelligent Systems for Molecular Biology*, *8*, 269–278.

Pietrokovski, S. (1996). Searching databases of conserved sequence regions by aligning protein multiple-alignments. *Nucleic Acids Research*, *24*(19), 3836–3845.

Rigoutsos, I., & Floratos, A. (1998). Combinatorial pattern discovery in biological sequences: The TEIRESIAS algorithm. *Bioinformatics*, *14*(1), 55–67.

Roepcke, S., Grossmann, S., Rahmann, S., & Vingron, M. (2005). T-Reg Comparator: An analysis tool for the comparison of position weight matrices. *Nucleic Acids Res*, *33*(Web Server issue), W438–441.

Saitou, N., & Nei, M. (1987). The neighbor-joining method: A new method for reconstructing phylogenetic trees. *Molecular Biology and Evolution*, *4*(4), 406–425.

Sandelin, A., Alkema, W., Engstrom, P., Wasserman, W. W., & Lenhard, B. (2004). JASPAR: An open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research*, *32*(Database issue), D91–94.

Sandelin, A., & Wasserman, W. W. (2004). Constrained binding site diversity within families of transcription factors enhances pattern discovery bioinformatics. *Journal of Molecular Biology*, *338*(2), 207–215.

Schneider, T. D., Stormo, G. D., Gold, L., & Ehrenfeucht, A. (1986). Information content of binding sites on nucleotide sequences. *Journal of Molecular Biology*, *188*(3), 415–431.

Schones, D. E., Sumazin, P., & Zhang, M. Q. (2005). Similarity of position frequency matrices for transcription factor binding sites. *Bioinformatics*, *21*(3), 307–313.

Sinha, S., & Tompa, M. (2002). Discovery of novel transcription factor binding sites by statistical overrepresentation. *Nucleic Acids Research*, *30*(24), 5549–5560.

Smith, T. F., & Waterman, M. S. (1981). Identification of common molecular subsequences. *Journal of Molecular Biology*, *147*(1), 195–197.

Wan, H., Li, L., Federhen, S., & Wootton, J. C. (2003). Discovering simple regions in biological sequences associated with scoring schemes. *Journal of Computational Biology*, *10*(2), 171–185.

Wang, H., Azuaje, F., & Black, N. (2004). Interactive GSOM-based approaches for improving biomedical pattern discovery and visualization. In J. Zhang, J. -H. He, & Y. Fu (Eds.), *Computational and information science: First international symposium, Proceedings*: *Vol. 3314/2004* (pp. 556–561). Springer-Verlag.

Wang, H. C., Badger, J., Kearney, P., & Li, M. (2001). Analysis of codon usage patterns of bacterial genomes using the self-organizing map. *Molecular Biology and Evolution*, *18*(5), 792–800.

Wang, T., & Stormo, G. D. (2003). Combining phylogenetic data with co-regulated genes to identify regulatory motifs. *Bioinformatics*, *19*(18), 2369–2380.

Wingender, E., Dietze, P., Karas, H., & Knuppel, R. (1996). TRANSFAC: A database on transcription factors and their DNA binding sites. *Nucleic Acids Research*, *24*(1), 238–241.

Workman, C. T., & Stormo, G. D. (2000). ANN-Spec: A method for discovering transcription factor binding sites with improved specificity. *Pacific Symposium on Biocomputing*, 467–478.

Zhao, X., Huang, H., & Speed, T. P. (2005). Finding short DNA motifs using permuted Markov models. *Journal of Computational Biology*, *12*(6), 894–906.