A Co-Evolutionary Framework for Regulatory Motif Discovery

Michael A. Lones, *Member, IEEE*, and Andy M. Tyrrell, *Senior Member, IEEE*
Intelligent Systems Research Group
Department of Electronics, University of York
Heslington, York, YO10 5DD, UK
mal503@ohm.york.ac.uk / amt@ohm.york.ac.uk

***Abstract*— In previous work, we have shown how an evolutionary algorithm with a clustered population can be used to concurrently discover multiple regulatory motifs present within the promoter sequences of co-expressed genes. In this paper, we extend the algorithm by co-evolving a population of Boolean classification rules in parallel with the motif population. Results using synthetic data suggest that this approach allows poorly conserved motifs to be identified in promoter sequences a magnitude longer than using population clustering alone, whilst results using muscle-specific data suggest the algorithm is able to evolve meaningful sequence classifiers in parallel with motifs.**

I. Introduction

Regulatory motifs describe the short patterns of nucleotides within DNA sequences which are responsible for controlling when and where a gene is expressed. More precisely, regulatory motifs describe transcription factor binding sites (TFBSs) embedded in the non-coding DNA sequences upstream, or more rarely downstream, of a gene's transcription start site (TSS). Within higher eukaryotes (such as *homo sapiens*), TFBSs are commonly found within the region extending several thousand kilobases upstream of the TSS, though the size of this promoter region varies widely and binding sites have been reported at distances in excess of 100kb from the gene they are regulating. TFBSs may also very rarely be found within the coding regions of genes. The typical length of a regulatory motif is 10–20bp [29] and binding sites for particular transcription factors can show substantial variation from gene to gene.

The aim of regulatory motif discovery is to identify the regulatory motifs responsible for a gene being expressed within a particular regulatory context [12], [20], [27]. This problem usually takes the following form: given a set of genes which are known to be expressed within a particular regulatory context, identify regulatory motifs which are over-represented in their promoter sequences relative to all other genes. Regulatory motif discovery is an important problem in contemporary bioinformatics, since mapping of gene regulatory regions underlies efforts to understand the gene expression networks which control development and metabolism.

The limitations of existing regulatory motif discovery tools have been highlighted in two recent comparative experimental studies [12], [27]. Some of the most notable problems are rapid failure with increasing sequence length and a tendency to discover biologically-meaningless patterns rather than true regulatory motifs. These are particularly problematic for higher eukaryotes, whose genes have long promoter sequences which often contain multiple, variable, TFBSs whose individual signals may be weaker than the background noise of spurious over-represented patterns.

Recently we have described an evolutionary algorithm which makes use of within-population data clustering to promote search for multiple diverse solutions within the same population [14]. This approach has proved successful both for finding multiple regulatory motifs within a set of sequences and for finding regulatory motifs in longer promoter sequences than can be handled by comparable statistical algorithms. In this paper, we present initial results looking at how co-evolution may be used to improve the search carried out by the evolutionary algorithm. In particular, we show how the co-evolution of Boolean rules describing relationships between motifs helps to guide the search for fit solutions, enabling the analysis of considerably longer promoter sequences than is possible using population clustering alone. We also discuss how this approach could be used to infer information about the structure of regulatory regions in higher eukaryotes.

The paper is organised as follows: Section II provides a summary of related work. Section III provides an overview of our earlier work using within-population data clustering. Section IV introduces the new co-evolutionary approach. Section V presents experimental results and analysis. Section VI discusses the implications and future directions of the work. Section VII concludes.

II. Related Work

There have been a number of previous studies in which evolutionary algorithms have been applied to regulatory motif discovery [3], [4], [6], [11]. Early work by Corne at al. [4] showed how consensus sequence strings and weight matrices could be evolved to describe core promoter motifs in the TSS-proximal region. Howard and Benson [11] evolved GP-Automata to describe motifs in 300bp sequences. Fogel et al. [6] used an island model distributed EA to discover regulatory motifs within the 1kb promoter regions of co-expressed genes. Congdon et al. [3] showed the ability of genetic algorithms to find conserved sequence strings in situations where exhaustive methods would be intractable. Evolutionary algorithms have also been applied to the problem of motif discovery in amino acid sequences. These and other biosequence applications are reviewed in [13].

For the majority of these approaches, the emphasis has been on applying fairly standard evolutionary algorithms to solve biosequence problems. This is motivated by the global, yet non-exhaustive, nature of search carried out by evolutionary algorithms: which can often lead to better solutions than the local and exhaustive search methods commonly used within bioinformatics. Our motivation is slightly different, in that we aim to make use of the relative flexibility with which evolutionary algorithms can represent and score solutions, and thereby develop approaches which are more applicable to solving motif discovery problems. For instance, in our earlier work [14] we focussed on the ability of evolutionary algorithms to represent diverse groups of candidate solutions concurrently, and showed that this mode of search has advantages both for discovering multiple motifs and for discovering single motifs in the presence of noise. In this work, we look at how co-evolution allows the motif discovery problem to be solved as a parallel combination of transcription factor binding site characterisation and sequence classifier induction.

Another significant difference between this work and the majority of other EA-based approaches [3], [6], [11] is that we focus upon the discovery of continuous (i.e. matrix) models of transcription factor binding sites rather than discrete (i.e. conserved strings and regular expressions) models. The continuous search space makes this a more difficult optimisation problem, yet a more suitable approach for characterising the variance within regulatory motifs.

III. Population Clustering

In [14], we described an evolutionary algorithm for regulatory motif discovery which applies data clustering to its population prior to the generation of new solutions. We refer to this population-based data clustering approach as *population clustering*. More precisely, our algorithm uses the sequential leader clustering algorithm [10] to partition the population into sub-populations prior to mating. Mating then takes place solely within sub-populations.

The sequential leader algorithm is a simple incremental clustering algorithm chosen primarily for its low time complexity. The algorithm makes a single pass through the population, and depending upon the degree of similarity, either assigns each solution to an existing cluster or uses it to seed a new cluster. The order-dependent nature of leader clustering also means that clusters can vary significantly from one generation to the next, providing an opportunity for solutions to move between clusters and thereby allowing a degree of genetic flow.

The explicit partitioning of the population is the primary advantage of population clustering over fitness sharing and crowding [21] and mating-based approaches [7] to diversity management—all of which achieve partitioning through indirect means—since it enables both selection and recombination to be carried out locally. In particular, selective pressure can be made high within partitions, promoting optimisation of individual solutions, whilst remaining low between partitions, promoting solution diversity. Unlike distributed populations [2], which also have explicit partitioning, this partitioning is determined by similarity between solutions rather than by evolutionary history, promoting better coverage of the search space. There have been several other examples of using population clustering within evolutionary algorithms, including [22] and [25].

In [14], this population clustering evolutionary algorithm (PCEA) was used to evolve variable length position frequency matrix (PFM) models of regulatory motifs. For a clustering metric, we used the Euclidean distance between the tetranucleotide distributions of two PFMs (following the approach of [8], in which this metric was successfully used to distinguish between families of regulatory motifs within the TRANSFAC [15] database). Our results showed this approach to be effective at both discovering multiple motifs concurrently and finding motifs in relatively long sequences.

IV. Motif-Rule Co-Evolution

In this paper, we extend this population clustering evolutionary algorithm by co-evolving Boolean rules that describe relationships between the evolving motifs. These Boolean rules have several potential roles: to identify combinations of motifs which can be used as sequence classifiers, to identify regulatory relationships between motifs, and, via selective feedback, to concentrate search within those motif clusters which contribute to fit rules. More generally, this is an example of cooperative co-evolution [17]: the intention being that fit motifs should enable the evolution of fit rules and fit rules should guide the evolution of fit motifs.

The algorithm is provided with two sets of sequences: a set of promoter sequences which are to be searched for over-represented motifs (the data set), and a set of sequences which capture the nucleotide background (the background set). The data set will usually consist of promoters from genes known to be co-expressed within a certain regulatory context, and the background set will typically comprise promoters from a larger random selection of genes, preferably known not to be expressed within the same regulatory context as those comprising the data set. The algorithm has two populations: a motif population and a rule population.

A. Motif Population

The motif population consists of position frequency matrices (PFMs), each describing a potential regulatory motif. The fitness of a PFM is a measure of how well it differentiates sequences in the data set from those in the background set. It is calculated as follows: Prior to fitness evaluation, the PFM is converted to a position weight matrix (PWM) for more efficient matching. This is done by translating frequencies into log-odds scores. For each sequence in both the data set and the background set, the best match to the PWM is found by calculating the PWM match score at each offset in the sequence. This value is then normalised to the range [0,1] by dividing by the maximum possible score for any PWM of equivalent size. The fitness of the motif is given by the difference between the mean best match score upon the data set and the mean best match score upon the background set.

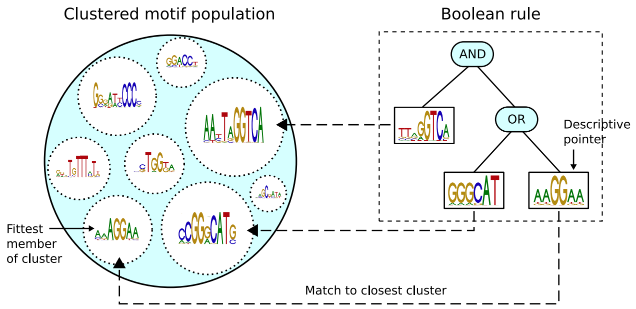


Fig. 1. Example of a Boolean classification rule, showing how descriptive pointers are matched to the fittest members of actual motif clusters. Position frequency matrices are drawn as sequence logos, which show the information content (column height) and distribution of nucleotide frequencies (letter heights) at each offset.

This is mapped linearly to a value between 0 and 1, with values above 0.5 indicating a better match against sequences in the data set than the background set.

The motif population is initially filled with randomly generated PFMs with uniformly distributed frequencies for each base. The motif population undergoes clustering at the beginning of each generation and new solutions are generated by within-cluster mating using both recombination and mutation. A fitness-ranked selection strategy is used to select parents for mating. Mutation is applied with a probability of 8% per nucleotide position and works by randomly changing the frequencies assigned to one or more residues and then normalising the other frequencies so that the total still sums to unity. Two forms of mutation are used. Gaussian mutation selects a new frequency value for a single residue using a Gaussian distribution centred around the current value and covering one standard deviation in the range ± 0.5. Values outside the range $[0, 1]$ are rejected. Gaussian mutation is applied during 90% of mutation events. A more disruptive mutation operator, which randomises then normalises all the residue frequencies in the selected matrix column, is applied during the remaining mutation events. There is also a 4% likelihood of adding a new column with random frequencies to either the start or end of a motif during mutation. Uniform crossover selects crossover points with a probability of 15% per nucleotide position, and then swaps the groups of matrix columns occurring between every other pair of crossover points. Mutation and crossover are applied independently in the ratio 7:3. Parameter settings were determined experimentally. The number of new solutions generated by each cluster is determined by a weighted function of the cluster's rank fitness and the mean fitness of rules which refer to the cluster (see section IV-C). To preserve diversity, each cluster generates at least one child solution.

B. Rule Population

The rule population consists of sequence classification rules, each defined by a Boolean expression. Rules are implemented as binary trees. Terminal nodes are pairs of the form (M, C) where M is a member of the motif population and C is a fractional cutoff value. For a given sequence S, a terminal node returns $true$ if the match value of M against S is equal to or greater than C multiplied by the highest possible match value of M. Non-terminal nodes are Boolean functions from the set $\{$AND, OR$\}$. The AND function expresses co-occurrence, i.e. both of its arguments return $true$ for each matching sequence. This could describe a regulatory context where, for example, two interacting transcription factors are required for a gene to be expressed. The OR function expresses alternative matches, i.e. at least one of its arguments returns $true$ for each matching sequence. This could capture a regulatory context in which more than one transcription factor could lead to expression, but only one is required.

Since motifs are undergoing evolution, and therefore subject to change and removal, rules can not refer directly to members of the motif population. Instead, they use descriptive pointers which are mapped to actual members of the motif population prior to evaluation of the rule (see Figure 1). In this implementation, these descriptive pointers are also position frequency matrices. Prior to evaluation, their tetranucleotide distribution is compared to that of the fittest member of each cluster within the motif population and they are mapped to whichever motif provides the closest match. This results in decoupling between the two populations, allowing them to vary independently whilst retaining behavioural links between rules and motifs.

The fitness of a rule reflects its ability to accept sequences in the data set whilst rejecting sequences in the background set. Raw fitness is calculated using Matthews correlation (MC), a measure of classifier accuracy defined by:

$$\frac{(TP \cdot TN) - (FP \cdot FN)}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \qquad (1)$$

where TP, TN, FP and FN are the numbers of true positive, true negative, false positive and false negative classifications respectively. A pseudocount is used to prevent zero values and the result is mapped to the range [0,1], where higher values are better and a value over 0.5 indicates a better-than-random classification. To avoid the problem of solution size bloat whilst still encouraging a degree of complexity, fitness penalties are imposed when tree size exceeds a certain depth and when the motifs referenced by a rule are not sufficiently diverse. A penalty of 0.04 is imposed for each level of depth above 5. The diversity penalty has two components. The first penalises multiple references to the same motif and is equal to a maximum penalty of 0.1 multiplied by the ratio of the number of motifs uniquely referenced by a rule to the total number of motifs referenced. The second penalises similarity between the referenced motifs and is equal to a maximum penalty of 0.05 multiplied by the ratio of the mean distance per base between the motifs' tetranucleotide distributions and the mean tetranucleotide probability for the motif population.

The initial rule population is filled with randomly generated binary trees with depths of between 2 and 5 levels. Cutoff values for terminal nodes have values of at least 0.7. Rules in subsequent populations are generated using

GP-like point mutation and sub-tree crossover. For point mutation, a single node is selected per tree. If the node is a terminal, there is a 55% probability of replacing its PFM with that of the fittest member of a randomly chosen cluster, a 20% probability of applying standard motif mutation to the existing PFM (see Section IV-A), a 10% probability of replacing the PFM with a randomly generated PFM, and a 15% probability of choosing a new cutoff value. Sub-tree crossover is slightly non-standard in that a crossover point is selected by first choosing a random depth and then selecting a random node at that depth. This is designed to reduce bloat by removing the bias towards leaf nodes associated with standard sub-tree crossover. Parent solutions are selected using tournament selection with a tournament size of 5.

C. Rule Feedback

Feedback from the rule population to the motif population occurs via the function which determines how many child solutions each cluster contributes to the next generation. This is calculated as follows: (1) the motif clusters are ordered by the relative fitness of their fittest member and assigned the corresponding motif fitness rank; (2) the motif clusters are ordered by the mean fitness of the rules which reference them and are assigned the corresponding rule fitness rank; (3) the motif clusters are ordered by the weighted mean of their motif and rule fitness ranks and assigned the corresponding overall rank; (4) each cluster is assigned a quota of child solutions linearly proportional to its overall rank. In cases where clusters are not referenced by any rules, they are assigned a rule rank one below the lowest position of those which are referenced. For the experiments reported in Section V, we use a ratio of 3:7 when calculating the weighted mean between motif and rule fitness ranks.

This procedure generates selective pressure towards PFMs with both a high individual match value and a high contribution to rule fitness. Nevertheless, motifs with a low individual match value will be ranked higher if they contribute to rule fitness. This provides a potential mechanism for identifying short or poorly conserved binding sites, whose individual signals can be impossible to distinguish from background noise. Since bound transcription factors often work in concert with those bound at other binding sites, these weak motifs could be recognised though their interactions with other motifs, which may be reflected in their contribution to rule fitness.

Our method of providing rule fitness feedback is somewhat different to most cooperative co-evolutionary algorithms, which typically provide feedback by directly modifying the fitness of those entities in the first population which are referenced by the second population. The advantage of our approach, which is made possible by the use of a clustered population, is that feedback affects groups of similar solutions in the first population rather than individual solutions. This should promote search within regions, rather than individual points, of the search space. It may also lessen the impact of the stochastic element of the process by which the second population samples members of the first population.

V. Results and Analysis

To determine the benefits of introducing rule co-evolution, we evaluated the extended algorithm using the same methodology used to evaluate the stand-alone population clustering EA [14]. The first experiment looked at the algorithm's ability to re-discover single known transcription factor binding sites embedded into DNA promoter sequences of various lengths. The aim of this experiment was to determine whether rule co-evolution affects the upper bound on the lengths of sequences which can be effectively searched. The second experiment applied the extended algorithm to the problem of rediscovering multiple known transcription factor binding sites embedded within promoter sequences. The aim was to determine whether co-evolution can identify multiple motifs (a task for which the stand-alone population clustering EA proved well-suited) and infer suitable classification rules. The third experiment applied the extended algorithm to a real biological data set comprising muscle-specific promoter sequences. The principle aim of this experiment was to determine whether co-evolved rules are able to express meaningful relationships between evolved motifs.

A. Rediscovering single motifs in synthetic data sets

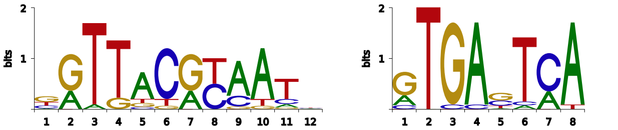


Fig. 2. Sequence logos for JASPAR motifs HLF (left) and c-FOS (right).

In previous work [14], we compared the ability of a population clustering evolutionary algorithm and two statistical motif discovery algorithms to locate motifs in synthetic sequences of various lengths. In this experiment, we followed this same approach but with considerably longer sequences. Data sets were constructed by embedding known regulatory motifs from the JASPAR [18] transcription factor binding site database at random locations within human upstream promoter sequences extracted from the Ensembl [1] genome database (release 43). Embedded motifs were generated probabilistically, with the probability of a particular nucleotide occurring at each offset in the motif directly proportional to its respective frequency in the corresponding JASPAR position frequency matrix. We used the JASPAR motifs HLF (ID: MA0043) and c-FOS (ID: MA0099), which have previously been shown to be relatively difficult to identify using motif discovery tools [5], [14]. This is due to poor conservation in the case of HLF and to short defining length in the case of c-FOS. Sequence logos for these motifs are shown in Figure 2. Data sets were generated for sequences of length 3kb, 5kb, 10kb and 20kb for both motifs. To make the problem more biologically realistic, motif instances were only embedded into half of the 100 sequences in each data set. Background sets consisted of 2000 randomly selected sequences. Data sets and background sets were non-intersecting. Motif and rule population sizes of 4000 were used in all runs with initial motif lengths of 5–50bp.

A total of 5 runs were carried out for each data set. This relatively small number of runs reflects the high computational overhead of processing long sequences. For example, with sequence length 20kb, a data set of 100 sequences, a background set of 2000 sequences, and a motif population of 4000, a single run of 100 generations takes approximately 6 days when distributed over 4 processors. This computation time is dominated by motif evaluation and we are currently looking at the potential use of hardware-based sequence matching to reduce evaluation time.

Table I shows the number of runs for each data set in which the embedded motif was successfully rediscovered by the algorithm. Table II compares these results ("Co-evolution") against those using the stand-alone population clustering evolutionary algorithm ("PCEA") [14] and the statistical algorithms MEME and NestedMICA (these figures are taken from [5]).

TABLE I

TOTAL RUNS (OUT OF 5) IN WHICH THE EMBEDDED MOTIF WAS SUCCESSFULLY REDISCOVERED

	Sequence length (base pairs)			
Motif	3000	5000	10000	20000
HLF	5	5	5	4
c-FOS	3	1	0	0

TABLE II

MAXIMUM SEQUENCE LENGTHS IN WHICH MOTIFS COULD BE CONSISTENTLY REDISCOVERED

	Algorithm			
Motif	MEME	NestedMICA	PCEA	Co-evolution
HLF	150	600	1500	20000
c-FOS	300	500	1500	3000

The interesting result here is that co-evolution allows the HLF motif to be found in sequences a magnitude longer (10–20kb) than those which could be effectively searched by the population clustering EA alone (1.5kb). A mechanism for this improvement is suggested by the rules inferred during the process of searching for the motif, an example of which is shown in Figure 3. Whilst individually the motifs referenced by this rule are not good characterisations of the HLF motif, the majority of them would have some probability of being generated by its PFM. For example, the well conserved 'TTACG' motif, labelled ① in Figure 3, would have a high probability of being generated by columns 3–7 in the PFM (see sequence logo in Figure 2). The motif 'A[AT]TTCG', labelled ②, would have a lower probability of being generated by columns 2–7 of the PFM, and 'TCATACG', labelled ③, would have a small probability of being generated by columns 5–11. In effect, the rule appears to group together the different components of the target PFM. The role of the functions AND and OR are less easy to see from this example, although it could be hypothesised that AND plays a role in gluing together horizontal components of the PFM (i.e. different column ranges), whilst OR presents choices between different vertical components (i.e. different row values). This suggests a pattern of search in which rule co-evolution 'identifies', and then constrains search towards, motif clusters representing the different components of a solution. Interactions between clusters (through movement of solutions and sequential clustering errors) may then lead towards composite solutions which more closely resemble the target motif.

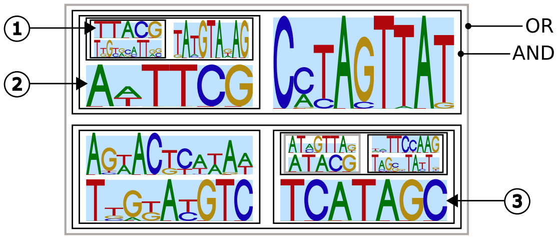


Fig. 3. The fittest Boolean rule in the final generation when searching for the HLF motif in sequences of length 10kb. For space efficiency, rules are drawn as tree-maps [23] in which functions appear as outlined boxes containing their arguments. Black boxes indicate AND and grey boxes indicate OR. Terminals are depicted by the sequence logo of the motif which they reference. The height of the blue box behind the sequence logo indicates the match cut-off value associated with the terminal.

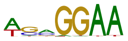
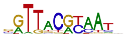
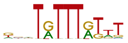
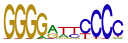
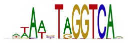
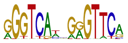
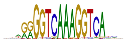
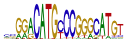
By comparison, rule co-evolution brings about a relatively modest improvement for the better conserved c-FOS motif—on the order of twice, rather than ten times, the sequence length. Assuming the above hypothesis is correct, this smaller improvement could be attributed to the lesser scope for vertical decomposition of the PFM, i.e. the c-FOS motif describes a relatively small number of alternative binding sites, presenting a smaller target for the hypothesised process of parallel exploration followed by consolidation. The motif's shorter defining length may also limit the potential for horizontal decomposition.

B. *Rediscovering multiple motifs in synthetic data sets*

In the second experiment, we applied the co-evolutionary algorithm to a synthetic data set containing 8 co-occuring JASPAR motifs. These motifs, listed in Table III, were chosen to cover a range of values for information content (IC) and defining length. In particular, RORA1, RXR-VDR and PPARG-RXRA were chosen in order to test the ability of a motif discovery algorithm to correctly distinguish between similar motifs, since these are all members of the nuclear receptor family and each contains the over-represented subsequence GGTCA. All of the motifs were stochastically inserted into 50 sequences of the 100 in the data set. A background set of 1000 sequences was used. Sequences in the data set and background set were both 1kb in length. A total of 5 runs were carried out.

TABLE III

JASPAR MOTIFS EMBEDDED IN MULTIPLE MOTIFS DATA SET.

#	Name	ID	IC	Length	Sequence Logo
1	SPI-B	MA0081	9.06	7	
2	HLF	MA0043	11.15	12	
3	FOXI1	MA0042	13.18	12	
4	NFKB1	MA0105	15.63	11	
5	RORA1	MA0072	17.42	14	
6	RXR-VDR	MA0074	20.45	15	
7	PPARG-RXRA	MA0065	23.45	20	
8	TP53	MA0106	26.24	20	

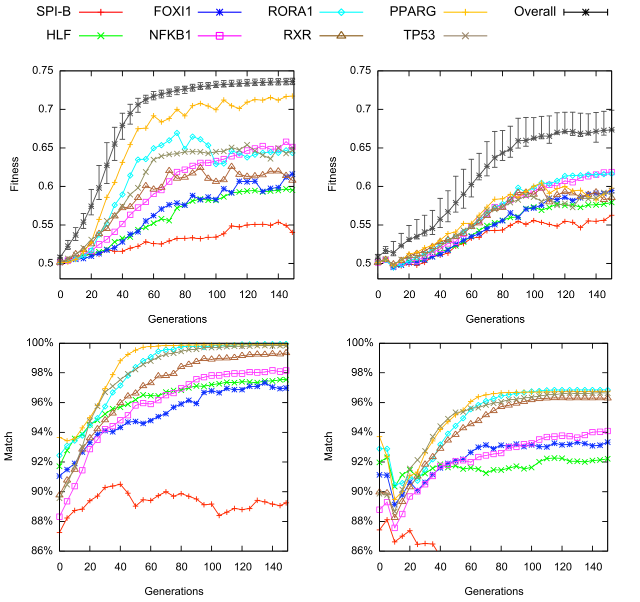


Fig. 4. Mean fitness (top) of closest matches (bottom) to the target motifs at each generation for the stand-alone population clustering EA (left) and the co-evolutionary algorithm (right). The mean fitness of the overall fittest motif present within the population is also shown. Match values were calculated using the dynamic programming technique described in [19].

Figure 4 shows the relative abilities of the stand-alone PCEA and the co-evolutionary algorithm to rediscover the embedded motifs. Whilst both algorithms are able to identify the majority of the embedded motifs, higher match and fitness scores indicate that the stand-alone PCEA is more able to optimise individual motif instances than the co-evolutionary approach. However, we did find that most runs of the co-evolutionary algorithm lead to optimal classification rules, i.e. rules which would accept all sequences containing the embedded motifs whilst rejecting those (in the data and background sets) that do not. This suggests that, for this problem at least, the search for fit rules interferes with the search for fit motifs. A likely reason for this, as illustrated by the optimal evolved rule shown in Figure 5, is that it is possible to correctly classify all sequences using only a subset of the embedded motifs—in this case NFKB1, TP53, the nuclear receptor motifs, and a small part of HLF—and furthermore, that it is not necessary to fully characterise the individual motifs.

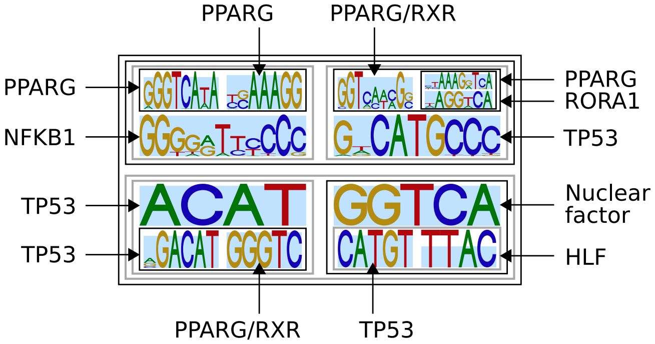


Fig. 5. Example of a rule which accepts all motif-containing sequences in the multiple motifs data set whilst rejecting all sequences in the data and background sets which do not contain the motifs.

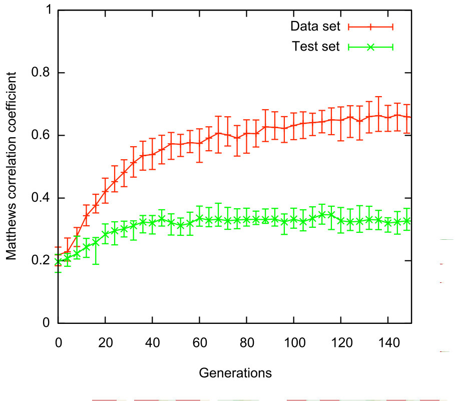


Fig. 6. Best classification of muscle-specific promoter sequences from non-muscle-specific promoter sequences at each generation. Points show mean values over 20 generations. Higher values indicate better classification accuracy. Values above zero indicate a better than random classification.

C. Discovering motifs in muscle-specific promoter sequences

In the final experiment, we looked at the co-evolutionary algorithm's ability to handle real biological data by attempting to infer classification rules from a set of muscle-specific promoter sequences. We used sets of promoter sequences described in [24][1]: a data set comprising Wasserman and Fickett's [28] curated collection of 43 muscle-specific promoter sequences; a background set comprising 2348 non-muscle promoter sequences from the eukaryotic promoter database (EPD); and, to measure the generality of evolved solutions, a test set comprising 28 muscle-specific promoter sequences from the EPD. Sequences in Wasserman and Fickett's dataset have lengths between 197bp and 802bp, those in the test set have lengths between 268bp and 600bp, and those in the background set have lengths between 91bp and 600bp. A total of 20 runs were carried out.

Figure 6 plots the evolution of classifier accuracy (in terms of Matthews correlation) over the course of 150 generations,

[1] Available from http://www.pnas.org/cgi/content/full/0406123102/DC1.

showing performance on both the data set and the test set. The best evolved classifiers in each run were able to reject almost all background sequences (all but one, on average) whilst accepting just over half (57% mean) of the sequences in the data set. The highest scoring classifiers on the test set also rejected most background sequences whilst accepting about a fifth (18% mean) of the positive examples. Whilst there is a clear disparity between performance on the data and test sets, these results do demonstrate that evolved classifiers are able to correctly classify sequences not seen during training whilst rejecting almost all background sequences.

Figure 7 shows the classifiers which performed best upon the data and test sets, respectively. Both contain motifs which resemble known muscle-specific transcription factor binding sites, including all those previously identified within Wasserman and Fickett's data set [28]. Motifs resembling binding sites for myocyte enhancer factor (MEF), serum response factor (SRF), transcription enhancer factor (TEF), and stimulating protein 1 (SP1) are present within both classifiers. Although less well characterised, motifs resembling myogenic determination factor (MyoD and Myf) binding sites can also be seen in each of the classifiers. It is also interesting to note that both rules make use of disjunctions (ORs) of conjunctions (ANDs), a form which allows subgroups of sequences to be classified separately. However, it is not clear whether the individual conjunctions represent alternative forms of single binding sites or capture composite regulatory modules consisting of multiple binding sites.

VI. Discussion

These initial results offer some interesting insights into the behaviour of the co-evolutionary algorithm. Perhaps most interesting, they suggest that rule co-evolution allows weakly conserved motifs to be discovered in long promoter sequences, something which is very significant given the fairly low sequence length limitations of many contemporary continuous-model (e.g. PFM) motif discovery algorithms [12]. Whilst the results also suggest that co-evolution does not appear to provide the same benefits for well conserved motifs, this is less significant given the relatively strong performance of existing discrete-model (e.g. consensus sequence) approaches to this sort of problem [27].

One point of concern is the potential for interference between motif evolution and rule evolution, as demonstrated by the multiple embedded motifs experiment. This is an example of a situation where the classification problem is easier than the motif discovery problem, and hence rule feedback does not exert sufficient selective pressure to push the motif population towards a diverse range of well characterised motifs. In situations such as this, it may be advisable not to use rule co-evolution. However, this problem is artificial in nature and is unlikely to reflect the difficulty of real promoter sequence classification problems, such as that addressed in the final experiment. The results from this muscle-specific sequence classification task show that rule co-evolution is able to induce classifiers containing known muscle-specific sequence features. These classifiers reject most non-muscle sequences and each accept a reasonable proportion of the muscle-specific sequences: and it seems plausible that a composite of multiple classifiers could improve this accuracy.

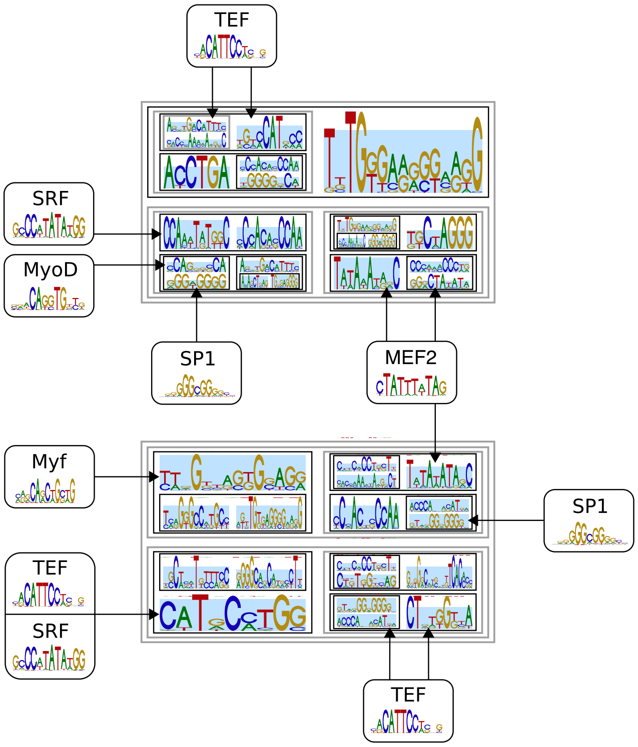


Fig. 7. Rules evolved to differentiate muscle-specific promoter sequences from non-muscle-specific promoter sequences. The best performing classifier (MC=0.806) upon the data set (top) correctly accepts 74% of sequences in the data set, 7% in the test set, and correctly rejects all but 3 of the background sequences. The best performing classifier (MC=0.458) upon the test set (bottom) correctly accepts 44% of sequences in the data set, 25% in the test set, and correctly rejects all but one of the background sequences. Both of these classifiers evolved in the same run. Labels show similarity to known muscle-specific transcription factor binding sites: SP1 (ID: M00196) and MyoD (M00001) in TRANSFAC, and MEF2 (MA0052), Myf (MA0055), SRF (MA0083) and TEF (MA0090) in JASPAR.

Nevertheless, there is substantial room for improvement, particularly with regard to classifier generality (i.e. performance on non-training sequences). Perhaps the most obvious next step is to improve the rule model used for classification, allowing rules to more accurately capture the structure of regulatory regions and thereby distinguish between positive and negative examples in a more meaningful fashion. In eukaryotes, it has been observed that transcription factor binding sites are often grouped together in cis-regulatory modules (CRMs). Usually these consist of 4–8 binding sites [29], including duplicates, and often there are functional constraints which limit the ordering, distances between, and strand orientation of the binding sites [26]. It would be quite feasible to introduce distance, ordering and strand occurrence constraints to the Boolean rule model, and possibly introduce a means of explicitly representing CRMs. This kind of approach has recently been shown to be effective for HMM-based approaches to motif discovery [16]. It would even be possible to introduce another layer of co-evolution to search for fit relationships between CRMs.

However, it is also important to not overly constrain the model, since the structure of regulatory regions does not always follow simple rules. In [9], for instance, it is argued that the commonly-held view of dense-cluster models of CRMs may be due to experimental bias towards yeast models, and does not reflect the complexity of regulatory regions in higher eukaryotes. In fact, given the relative flexibility with which solutions can be represented in evolutionary algorithms, approaches such as the one described in this paper could play a role in improving understanding of the organisation of regulatory regions.

VII. Conclusions

In this paper we have presented initial results from our investigation of a co-evolutionary architecture for regulatory motif discovery. Our approach builds upon previous work in which we developed a population clustering evolutionary algorithm designed to concurrently discover multiple, diverse over-represented patterns present within the promoter sequences of co-regulated genes. This new work introduces a co-evolutionary layer in which sequence classification rules are evolved in parallel with motifs, providing feedback to guide the evolution of the motif population. We have applied this new algorithm to several different problems involving synthetic and real biological data. The results suggest that rule co-evolution considerably improves the ability of the algorithm to discover weakly conserved motifs in long promoter sequences (10–20kb). Analysis of evolved classifiers indicates that this improvement may be a result of implicit problem decomposition carried out during rule evolution. However, there is some concern that motif evolution will be impaired if the classification problem is easier than the motif discovery problem. The results also show that the algorithm is able to evolve meaningful classifiers when applied to a real biological data set. Nevertheless, the limited generality of the evolved classifiers suggests that more constraints should be introduced to the classifier model in future work.

Acknowledgements

We would like to acknowledge the support of the White Rose Grid in providing computational resources for this research project.

References

[1] E. Birney, T. D. Andrews, P. Bevan, et al. An overview of ensembl. *Genome Res*, 14(5):925–928, May 2004.

[2] E. Cantú-Paz. *Designing Efficient and Accurate Parallel Genetic Algorithms*. Ph.D. thesis, University of Illinois at Urbana-Champaign, USA, 1999.

[3] C. B. Congdon, C. Fizer, N. W. Smith, et al. Preliminary results for GAMI: A genetic algorithms approach to motif inference. In *Proc. 2005 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, CIBCB 2005*, pp. 97–104. IEEE, 2005.

[4] D. Corne, A. Meade, and R. Sibly. Evolving core promoter signal motifs. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001*, pp. 1162–1169. IEEE Press, 27-30 May 2001.

[5] T. A. Down and T. J. P. Hubbard. NestedMICA: sensitive inference of over-represented motifs in nucleic acid sequence. *Nucleic Acids Res*, 33(5):1445–53, 2005.

[6] G. Fogel, D. Weekes, G. Varga, et al. Discovery of sequence motifs related to coexpression of genes using evolutionary computation. *Nucleic Acids Research*, 32(13):3826–3835, 2004.

[7] R. Fry, S. Smith, and A. Tyrrell. A self-adaptive mate selection model for genetic programming. In *Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2005)*, vol. 3, pp. 2707–2714. IEEE Press, 2005.

[8] K. Grote, R. Schneider, and T. Werner. Kohonen maps are suitable for a biologically meaningful classification of transcription factor binding site matrices. In *Proceedings of German Conference on Bioinformatics (GCB '99)*. 1999.

[9] M. Halfon. (Re)modeling the transcriptional enhancer. *Nature*, 38:1102–1103, 2006.

[10] J. Hartigan. *Clustering Algorithms*. John Wiley & Sons, Inc., New York, 1975.

[11] D. Howard and K. Benson. Evolutionary computation method for pattern recognition of cis-acting sites. *Biosystems*, 72(1-2):19–27, Nov. 2003.

[12] J. Hu, B. Li, and D. Kihara. Limitations and potentials of current motif discovery algorithms. *Nucleic Acids Res*, 33(15):4899–4913, 2005.

[13] M. A. Lones and A. M. Tyrrell. The evolutionary computation approach to motif discovery in biological sequences. In F. Rothlauf, ed., *Genetic and Evolutionary Computation Conference (GECCO) 2005 Workshop Program, Workshop on Biological Applications of Genetic and Evolutionary Computation*, pp. 1–11. ACM, June 2005.

[14] M. A. Lones and A. M. Tyrrell. Regulatory motif discovery using a population clustering evolutionary algorithm. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, published online January 2007.

[15] V. Matys, E. Fricke, R. Geffers, et al. TRANSFAC: transcriptional regulation, from patterns to profiles. *Nucleic Acids Res*, 31(1):374–8, Jan 2003.

[16] K. Noto and M. Craven. Learning probabilistic models of cis-regulatory modules that represent logical and spatial aspects. *Bioinformatics*, 23(2):e156–e162, Jan 2007.

[17] M. A. Potter and K. A. D. Jong. Cooperative coevolution: An architecture for evolving coadapted subcomponents. *Evolutionary Computation*, 8(1):1–29, March 2000.

[18] A. Sandelin, W. Alkema, P. Engström, et al. JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res*, 32:D91–4, Jan 2004.

[19] A. Sandelin, A. Höglund, B. Lenhard, et al. Integrated analysis of yeast regulatory sequences for biologically linked clusters of genes. *Funct Integr Genomics*, 3(3):125–134, Jul 2003.

[20] G. K. Sandve and F. Drabløs. A survey of motif discovery methods in an integrated framework. *Biol Direct*, 1:11, 2006.

[21] B. Sareni and L. Krähenbühl. Fitness sharing and niching methods revisited. *IEEE Transactions on Evolutionary Computation*, 2:97–106, 1998.

[22] L. Schnitman and T. Yoneyama. A clustering method for improving the global search capability of genetic algorithms. In F. M. G. França and C. H. C. Ribeiro, eds., *Proc. 6th Brazilian Symposium on Neural Networks (SBRN 2000)*, pp. 32–37. IEEE Computer Society, 2000.

[23] B. Shneiderman. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on Graphics*, 11(1):92–99, January 1992.

[24] A. D. Smith, P. Sumazin, and M. Q. Zhang. Identifying tissue-selective transcription factor binding sites in vertebrate promoters. *Proc Natl Acad Sci U S A*, 102(5):1560–5, Feb 2005.

[25] F. Streichert, G. Stein, H. Ulmer, et al. A clustering based niching method for evolutionary algorithms. In E. Cantú-Paz, J. A. Foster, K. Deb, et al., eds., *Proc. Genetic and Evolutionary Computation Conference (GECCO 2003)*, vol. 2723 of *LNCS*, pp. 644–645. Springer-Verlag, Chicago, 12-16 July 2003.

[26] G. Terai and T. Takagi. Predicting rules on organization of cis-regulatory elements, taking the order of elements into account. *Bioinformatics*, 20(7):1119–1128, May 2004.

[27] M. Tompa, N. Li, T. L. Bailey, et al. Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotechnol*, 23(1):137–44, Jan 2005.

[28] W. W. Wasserman and J. W. Fickett. Identification of regulatory regions which confer muscle-specific gene expression. *J Mol Biol*, 278(1):167–81, Apr 1998.

[29] G. A. Wray, M. W. Hahn, E. Abouheif, et al. The evolution of transcriptional regulation in eukaryotes. *Molecular Biology and Evolution*, 20(9):1377–419, Sep 2003.