Trade-offs Using GAMID For The Inference of DNA Motifs That Are Represented in Only a Subset of Sequences of Interest

Jeffrey A. Thompson
University of Southern Maine
Portland, ME 04104, USA
jeffrey.ahearn.thompson@maine.edu

ABSTRACT

In prior work, we presented GAMID, an extension of GAMI (Genetic Algorithms for Motif Inference), which allows the system to ignore some of the sequences when looking for candidate conserved motifs in noncoding DNA. This ability is useful both when looking for candidate motifs in co-expressed genes (where it is not expected that all genes respond to the same transcription factors) and when looking for candidate motifs in divergent species (where functional elements might appear only in related species). In these cases, we would like to allow the inferred motif to be present in only a subset of the input data. By excluding some sequences from the match process, GAMID succeeded at finding known functional elements. Here we use the results of experiments using artificial data with GAMID to show that GAMID's success in inferring motifs in subsets of the input data results in it finding fewer motifs when they are present in all the sequences. Therefore, GAMID is useful as an adjunct tool to GAMI, but is not a replacement for its functionality.

Categories and Subject Descriptors

J.3 [**Life and Medical Sciences**]: Biology and Genetics; I.2.8 [**Artificial Intelligence**]: Problem solving, Control methods, and Search

General Terms

Algorithms

Keywords

Evolutionary computation, genetic algorithms, DNA motif inference

1. INTRODUCTION

GAMI [7, 8] (Genetic Algorithms for Motif Inference) uses a Genetic Algorithms (GA) search to identify putative functional elements in noncoding DNA. The system was designed to identify putative functional elements following the notion that elements that have been conserved across evolution are more likely to be functional; therefore, GAMI seeks to find highly conserved patterns in the data, which are called motifs. In previous work, GAMI has been shown to be adept at finding highly conserved elements in long sequence lengths (e.g., 100kb) and across several dozen sequences.

It is thought that co-expressed genes frequently share regulatory elements, so it should also be possible to use GAMI for the inference of motifs in co-expressed genes. However, a limitation of GAMI in this regard is that it seeks to find evidence of a motif in all sequences of the data, and this is not appropriate when investigating genes that may be co-regulated. Even if the target genes share some functional regulatory elements, it is unlikely they share them all. Often, a functional element will appear in only a subset of the sequences [1], [16]. Similarly, in a dataset of highly divergent species, it is likely that some species have developed novel methods of regulation for what are still orthologous genes [6]. Therefore, a functional element might be present in sequences from closely related species, but not appear in all sequences in the dataset.

Therefore, we developed GAMID [19], an extension to GAMI with the capability of selectively ignoring sequences when evaluating motifs if it seems likely that doing so will be beneficial in identifying a putative functional element. We refer to this process as "dropout". In prior work we showed that GAMID was able to find known functional elements present in only a subset of the input sequences. However, we did not establish if GAMID can function as a replacement for GAMI, or if the process of identifying motifs in subsets of the input negatively affects its ability to identify better represented motifs. In this work, we study how our changes to GAMI affect GAMID's ability to identify well represented motifs.

In this paper, we first present background information on genetic algorithms and the problem of motif inference as it relates to both co-expressed and orthologous genes, including the particular problem of motifs represented in only a subset of the sequences of interest. Second, we discuss GAMID, our solution to that problem and how it relates to this current work. Third we discuss our methodology. Fourth, we analyze our results and show that there appears to be a trade-off in the results when using GAMID. Finally, conclusions are presented and future work is considered.

GECCO'12 Companion, July 7–11, 2012, Philadelphia, PA, USA.

2. BACKGROUND

2.1 Motif Inference in Orthologous Genes

Several studies suggest that comparative analysis, when applied to evolutionarily diverse organisms, helps to predict functionally important noncoding regions [9], [18], [10]. Motif inference in this case involves analyzing the regulatory regions of orthologous genes to look for areas of high conservation. The hypothesis motivating this approach is that highly conserved regions are more likely to contain functional elements, and that is why the regions have been conserved through evolution [11], [23].

2.2 Approaches to Motif Inference

Depending on the number of sequences being examined, the sequence length, and the motif length, exhaustive searches can be prohibitively expensive. Therefore, most approaches to motif inference use some sort of heuristic search technique.

As noted by Lones and Tyrell [15], the most common approach to locating and characterizing conserved regions in sets of biological sequences is to first use a global sequence-alignment system. However, global sequence alignment is computationally expensive, particularly as the number and length of the sequences increases. Furthermore, for evolutionarily distant species, the degree of sequence divergence precludes global alignment, especially in noncoding regions, which often exhibit less overall conservation than coding regions.

Another highly favored approach to motif inference is to search for an optimized and coadapted set of window locations across the set of sequences. The set of windows form a matrix that describes the motif. This approach is used, for example, in Multiple Expectation Maximization for Motif Elicitation (MEME) [2], Gibbs Sampler [20], and Fogel et al. [12]. The latter also uses an evolutionary computation approach. Note that these approaches are also suited to detection of overrepresented motifs in co-expressed genes.

2.3 GAMI's Approach to Motif Inference

GAMI was designed to infer motifs using sequences from evolutionarily distant species [7, 8]. Of course, the specific goals of motif inference vary for different researchers. With GAMI, the goal was to find a more computationally tractable approach than those listed above; therefore, it searches the space of possible motifs instead of the space of possible matrices, and it does not rely on global alignment. The following characteristics describe the unique combination of requirements for GAMI:

1. GAMI was originally designed to look for conserved regions using data sets with support for the motifs identified in each of the sequences in the set. The work on GAMID changed this requirement.

2. Searches are done in noncoding regions, where there is usually less overall conservation than in coding regions.

3. Searches are for putative regulatory regions and not specifically transcription factor binding sites.

4. The ability to search in long sequence lengths, perhaps, 100 kb or longer.

5. The ability to search a large number of sequences when quality sequence is available.

6. Computationally tractable.

Several of these characteristics differ from other motif inference approaches. For example, many motif inference approaches do not require that motifs are contained in all of the sequences; this is evidenced in published benchmarking data sets such as [21]. Some tools search only for TFBSs and are therefore limited in the scope of regulatory elements that can be identified [4]. Some motif inference projects look in the core promoter region only (for example, 1-200 bp). Many motif inference projects restrict the input to a small number of sequences or short sequences due, in part, to runtime concerns. GAMI's use of Genetic Algorithms to search the possible motif space means that it is not constrained by most of these limitations.

For the purposes of this work, a motif is defined in a given data set of nucleotide sequences as an N-mer that occurs in the sequences being examined. We allow imperfect matches, so that the pattern might not be represented exactly in one or more of the input sequences. N-mers that are more strongly matched across the set of sequences are considered stronger motifs.

2.4 Finding Overrepresented Motifs in Co-expressed Genes

The idea behind these motif inference systems can be extended to the study of co-expressed genes. The difference is that in this case motifs are not a product of conservation through evolution. Instead, it is assumed that genes that are co-expressed are often regulated by a common transcription factor. Therefore, these genes are assumed to share TFBSs more often than other genes would by chance [1], [16]. Therefore, motifs for these binding sites should be overrepresented in the input sequences, compared to what would be found by chance. A number of methods have been developed to aid in the detection of such overrepesented motifs, including MEME [2], Gibbs Sampler [20], Fogel et al [12], Frith et al. [13], and Zheng et al. [24]. For the most part, these approaches use various statistical methods to find motifs that are overrepresented in the input data, often by comparing motifs to some set of control data, such as a random sample of genes ([12] is an exception to this). In addition, some methods of statistical inference for overrepresented motifs also make use of phylogenetic information to improve the accuracy of their results, such as oPOSSUM [14], PhyloCon [22], and BlockSampler/BlockAligner [16]. These methods first identify putative functional elements in each input sequence using phylogenetic footprinting, then use statistical analysis to identify common elements. This limits the number of motifs that must be considered, which is computationally more tractable and may help eliminate false positives.

2.5 Identifying Motifs Represented in Subsets of Gene Sequences

As we have discussed, when examining the regulatory regions of both co-expressed and orthologous genes, a subset of the input sequences may be missing motifs that are present in the others. Actually, there may be numerous subsets, each of which share certain motifs. Therefore, to identify motifs that are represented only in a subset of the input, it

is necessary to have some method of comparison that will not be overly disrupted by sequences missing some motifs.

There are a number of different approaches described in the literature, each of which has strengths and weaknesses. We have tried to make the following examples representative of the most popular:

1. MEME is a widely used approach that discovers motifs through a statistical process known as *expectation maximization*. Essentially, this involves a measure of the probability of each letter occuring at any position in a pattern. This is achieved in part through local alignments between sequences to find recurring patterns, though there is also the option to use background data to improve the sensitivity of the results. MEME is able to incorporate sequences missing motifs into this process, though it obviously does interfere, to an extent, with its probability calculations. According to the authors [2], MEME is better suited to input sequences less than 1000 bp long, with a minimal number of input sequences that do not share motifs.

2. Blanchette et al. [3], developed a statistical method for motif discovery, known as Footprinter, that incorporates phylogenetic information to find *parsimony scores*. This involves finding the number of changes it takes when looking along a phylogenetic tree in order to move from one form of a motif to another. It then identifies the motifs with the smallest number of changes (i.e. the most parsimonious). It is also able to handle motifs that occur in only a subset of the input sequences. Footprinter will show a motif as being conserved if it has a low parsimony when considering how diverged the species are that the subset of sequences are from. This can be a useful approach for the discovery of regulatory elements when a researcher has sufficient orthologous sequence with which to test, and is able to construct a phylogenetic tree that shows the evolutionary relationships between the input sequences. However, this is not always the case. Also, FootPrinter is computationally expensive, and problems involving large motifs or long sequences can be intractable with this approach. It is also important to note that FootPrinter was specifically designed for phylogenetic footprinting, not the study of co-expressed genes.

3. A number of other approaches, including methods using evolutionary computation, are able to handle some sequences missing motifs present in the others, but few appear to be designed with that functionality in mind. GALF-P was designed to handle such missing motifs [5]. However, this is handled in a post-processing step. Therefore, solutions must survive to the final population to be considered, which may hide motifs that are strong only in the context of a subset.

In general, statistical methods are computationally intensive, which may limit the sequence length that can reasonably be investigated. Depending on the data set, motifs may be too long to handle with such approaches. Meanwhile, methods depending on phylogenetic trees work well in some circumstances, but are not applicable to working with co-expressed genes. Multi-step approaches that combine orthologous and co-expressed genes also depend on a greater availability of data, making some data sets difficult or impossible to work with. Although systems for working with motifs represented only in a subset of the input do exist, most were not designed with that functionality in mind, and we found none that met our requirements for working with long sequences, long motifs, and a large number of input sequences.

```
PSG3   GAGGGGACAGAGAGGTGTCCTGGGCCTGACCCCGCC 17/17
PSG5   GAGGGGACAGAGAGGTGTCCTGGGCCTGACCCCACC 17/17
PSG7   TGTCCTGGGCCTGACCCCACCCATGAGCTTGAGAAG 17/17
PSG9   TGTCCTGGGACTGACCCCGCCCATGAGCTTGAGAAG 16/17
PAPPA  TGTCCAGGACATCTGCCTTTCAGAAGCTGTAGTCCT 11/17
CSH1   TCAGAACCCCCACAATCTATTGGCTGTGCTTGGCCC 11/17
CSH2   TGACCTGGGGGAGCCCCACCGCCTCCGCCCCAAGGT 12/17
                                  Total: 101/119
                                         (84.88%)
```

Figure 1: An example of the 17-mer motif TGTCCTGGGCCTGACCC matching locations in short selections of sequences from a few pregnancy-related genes. The motif location is shown in bold type. Locations where a sequence's best match differs from the consensus motif are shown in red. The overall Match Percentage (MP) score for this motif is 84.88 %, which corresponds to matching 101 of the possible 119 bases (17 bases in each of 7 sequences).

3. GAMID

In order to meet the objective of finding motifs represented in a subset of the input data while at the same time being able to handle a large number of long sequences and possibly long motifs, we decided that it made sense to extend GAMI. GAMI has already performed well on motif inference without dependence on phylogenetic trees, or computationally expensive statistics. Because our solution to this problem involves allowing GAMI to selectively ignore sequences during motif evaluation, we refer to this approach as GAMI with Dropout (GAMID).

GAMI was originally designed to search a set of nucleotide sequences for patterns that appear at least once in each sequence. The motif representation is the standard consensus motif: an N-mer composed of the bases A, C, G, and T. For example, if we are searching for 8-mers, possible motifs identified would include `CATGCAAT`, `TAGGAACT`, `ACTTACGT`, and so forth.

The fitness function uses a metric called "match percent" (MP). To evaluate the MP of a given motif, each sequence is searched to find the best consecutive match for that motif within that sequence. Forward and reverse-complement matches are considered for each sequence. The best match maximizes the number of bases that match the motif across all the sequences; there might be more than one best match for a given motif and nucleotide sequence (but this does not alter the score). An example match for the motif `TGTCCTGGGCCTGACCC` is shown in Fig. 1. The (maximum) number of bases matched in each sequence is the score for that motif with that sequence. The score for the motif across all sequences in the data is the percent of the overall matches found out of the theoretical maximum possible (number of input sequences $\times$ motif length).

Our modifications to GAMI involved giving it the ability to allow a sequence to "dropout" from the evaluation process when scoring a motif as described above. In this case, the MP score of a motif reflects the percentage of bases matched in only a subset of the sequences examined.

GAMID accomplishes this in the following manner:

- In GAMI's fitness function, each time the best match for a motif is found, GAMID adds the amount that match contributes to a motif's fitness to an array, indexed by sequence number.
- Once the motif's total fitness is calculated, GAMID traverses the array and checks if the total percent of matches would be better if the sequence was excluded from the motif's fitness.
- If dropping the sequence would improve the motif's score by a user-defined threshold value, then the sequence number is added to the motif's internal list of dropped sequences.
- Finally, the motif's fitness is calculated based on which sequences are retained.

It is worth noting that if the Dropout Threshold is set too low (i.e. it is very easy for a sequence to drop), then GAMID will simply drop all but one sequence. In that case, GAMID will report all motifs as perfectly conserved, since they will not be compared against any other sequences. Therefore, it is important to find an appropriate Dropout Threshold setting that reveals previously hidden subsets that share motifs, without allowing too many sequences to drop. Furthermore, since conservation between sequences will vary by data set, the most appropriate setting will probably have to be found through experimentation.

4. METHODOLOGY

In prior work, it has been establed that GAMID is able to find known functional elements in biological sequences even when the elements are present in only a subset of the input data. Now we would like to determine if GAMID is also able to identify better represented motifs as well as the original GAMI, or if there is some trade-off in its approach.

To that end, we ran GAMI and GAMID using artificial DNA sequences containing implanted motifs using a variety of DropoutThreshold settings. By comparing the recovery rates of both approaches using motifs that are either well or poorly represented in the input, we should be able to show if GAMID is able to recover well represented motifs for settings at which is also recovers a maximal number of poorly represented motifs.

4.1 Data

We generated artificial DNA sequences using Rouchka and Hardin's rMotifGen V2.0 [17]. We created 100 sets of 10 sequences with 10 implanted motifs using the following settings:

- 5 of the motifs were present in every sequence, but were 90% degraded. In other words, each implanted motif was 90% similar to the original one generated by rMotifGen. This was done to simulate a more realistic motif problem, in which conserved motifs are seldom perfect matches.
- 5 of the motifs were present in only 6 of the 10 sequences (these motifs were also 90% degraded). This setting was chosen to make the problem more challenging, to make any differences between GAMI and GAMID more apparent.
- The ACGT content of each motif was varied. For each set of motifs, the following percentages of each base were used (in the order A,C,G,T). (Motifs 1-5 appear in each sequence, Motifs 6-10 appear in only 6 out of every 10 sequences):
  - Motif 1: 25,25,25,25
  - Motif 2: 50,30,15,5
  - Motif 3: 5,50,30,15
  - Motif 4: 15,5,50,30
  - Motif 5: 30,15,5,50
  - Motif 6: 25,25,25,25
  - Motif 7: 50,30,15,5
  - Motif 8: 5,50,30,15
  - Motif 9: 15,5,50,30
  - Motif 10: 30,15,5,50

4.2 Parameter Settings

In this work we are comparing our modifications of GAMI to its original form. Therefore, for the time being, the most important factor is that the settings are consistent between the two versions. For all experiments reported here, we used a population size of 1,000, crossover rate 0.8, and a mutation rate of 0.02. We used a motif length of 15, performing 20 runs for each data set. The length of motif was chosen to make the problem challenging, again with the hope of making differences between GAMI and GAMID runs more apparent.

For each run, 55,000 trials were performed. The number of trials is the number of fitness function evaluations. Due to elitism, and the ability to recognize when a reproduction operator has no effect, there is not a clean mapping between trials and the number of generations.

In order to elucidate the affect of GAMID on our results, we ran it against all 100 data sets using a range of 20 different DropoutThreshold settings, starting at .001 and working up to .020 in increments of .001. A setting of .020 means that a motif would need to be 2% stronger than it would have been, in order to justify dropping a sequence.

Fifty percent elitism was used to preserve the best 500 motifs in the population every generation. Thus, at most 500 new motifs are created each generation, and the "result" of a run can be considered to be the 500 best solutions in the final population. The 80 percent crossover rate means that 80 percent of the remaining motifs are candidates for crossover (a total of 150). The 2 percent mutation rate means that each nucleotide in a solution has a 2 percent chance of being set to a random value (possibly, the same as it was before). Rank-based selection was used.

5. RESULTS

As stated above, there were 20 runs performed for each of 100 data sets at each of 20 DropoutThreshold settings for GAMID, as well as another 20 runs for each 100 data sets with GAMI. Due to the stochastic nature of Genetic

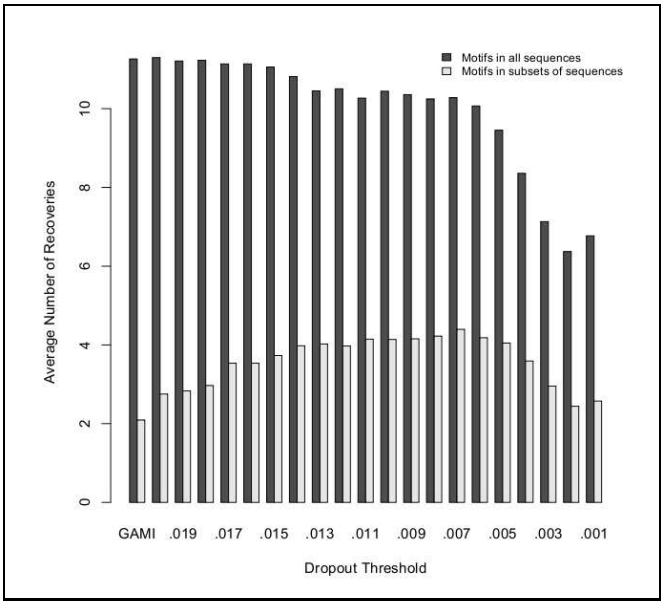

Figure 2: Average number of recoveries using GAMI and GAMID at different DropoutThresholds. The black bars correlate to the motifs present in all sequences. The gray bars are recoveries of motifs present in subsets of sequences. GAMID recovers the greatest number of motifs present in subsets between DropoutThresholds of .006-.008, but not as many strongly represented motifs.

Algorithms, it is to be expected that not all motifs will be recovered in any particular run, especially with a difficult problem. We decided to count how many runs a motif was recovered in across all 100 data sets for GAMI and for each DropoutThreshold setting used for GAMID. Since this left us with 21 (setting) $\times$ 100 (data sets) $\times$ 10 (motifs) = 21000 results, we decided to average the results for each motif. Figure 2 shows the average recovery rates across the 21 different settings (1 GAMI, 20 GAMID), grouped by motifs present in every sequence (shown in black) and motifs present in a subset of sequences (shown in gray). Table 1 shows the average recovery rates of GAMI and GAMID (using Dropout Thresholds of .006-.008) for motifs that appear in all sequences and those that appear in only a subset.

6. DISCUSSION

In Figure 2 a couple of things can be seen:

1. When recovery of the motifs that appear in only a subset of the data is best (using DropoutThreshold settings of .006-.008), the recovery of well represented motifs is lower than it was for GAMI.

2. At extremely low DropoutThreshold settings, the implanted motifs are recovered much less frequently, both for those well represented and those that appear only in a subset of the sequences. This shows that it is probably not beneficial to use too lax a setting for this parameter.

In Table 1 it can be seen that for DropoutThresholds .006-.008, GAMI's results were on average stronger by a factor of 1.10 than GAMID for motifs present in each sequence. However, GAMID's results were on average stronger by a factor of 2.01 for motifs present in only a subset of the sequences. For these data, GAMID recovered fewer well represented motifs when it recovered the most poorly represented ones.

Table 1: Average Recovery Rates for GAMI vs GAMID at Dropout Thresholds .006-.008

	GAMI	.006	.007	.008
Motifs in all seq's	11.26	10.07	10.28	10.25
p-value	-	0.0001	0.0015	0.0007
Motifs in subsets	2.09	4.18	4.40	4.22
p-value	-	< 0.0001	< 0.0001	< 0.0001

7. CONCLUSIONS

The goal of this work was to evaluate how adding "dropout" functionality to the motif inference software GAMI affected its ability to recover motifs that are well represented in the input sequences. GAMID has been shown to identify functional motifs that are represented in only a subset of the input sequences. With the artificial sequences used in this work GAMID was able to recover such motifs even when given a challenging problem. Nevertheless, our results show that GAMI is better at recovering well represented motifs present in the same data set. This suggests that GAMID is not currently a replacement for GAMI, but instead, GAMID should be viewed as a useful alternative to GAMI for particular problems.

8. FUTURE WORK

In future work we would like to identify exactly why the trade-off discovered with GAMID exsits. The number of motifs used in this work would not preclude GAMID from having recovered all the motifs, so perhaps there is some change that we can make to the algorithm that would enable it to do so.

Acknowledgements

This project was supported by a National Science Foundation (NSF) CAREER award (#953495), NSF Cooperative Agreement No. HRD-0833567, grants from the National Center for Research Resources (5P20RR024475-02) and the National Institute of General Medical Sciences (8 P20 GM103534-02) from the National Institutes of Health, and Maine Economic Improvement Funds.

Thanks to Dr. Clare Bates Congdon for her advice and encouragement on this project.

9. REFERENCES

[1] D. J. Allocco, I. S. Kohane, and A. J. Butte. Quantifying the relationship between co-expression, co-regulation and gene function. *BMC Bioinformatics*, 5:18, Feb 2004.

[2] T. L. Bailey, N. Williams, C. Misleh, and W. W. Li. MEME: discovering and analyzing DNA and protein sequence motifs. *Nucleic Acids Res.*, 34:W369–373, Jul 2006.

[3] M. Blanchette and M. Tompa. FootPrinter: A program designed for phylogenetic footprinting. *Nucleic Acids Res.*, 31:3840–3842, Jul 2003.

[4] K. Cartharius, K. Frech, K. Grote, B. Klocke, M. Haltmeier, A. Klingenhoff, M. Frisch, M. Bayerlein, and T. Werner. MatInspector and beyond: promoter analysis based on transcription factor binding sites. *Bioinformatics*, 21:2933–2942, Jul 2005.

[5] T. M. Chan, K. S. Leung, and K. H. Lee. TFBS identification based on genetic algorithm with combined representations and adaptive post-processing. *Bioinformatics*, 24:341–349, Feb 2008.

[6] H. Chiba, R. Yamashita, K. Kinoshita, and K. Nakai. Weak correlation between sequence conservation in promoter regions and in protein-coding regions of human-mouse orthologous gene pairs. *BMC Genomics*, 9:152, 2008.

[7] C. B. Congdon, J. C. Aman, G. M. Nava, H. R. Gaskins, and C. J. Mattingly. An evaluation of information content as a metric for the inference of putative conserved noncoding regions in DNA sequences using a genetic algorithms approach. *IEEE/ACM Trans Comput Biol Bioinform*, 5:1–14, 2008.

[8] C. B. Congdon, C. Fizer, N. W. Smith, H. R. Gaskins, J. C. Aman, G. M. Nava, and C. J. Mattingly. Preliminary results for gami: A genetic algorithms approach to motif inference. In *CIBCB'05*, pages 97–104, 2005.

[9] G. M. Cooper, M. Brudno, E. D. Green, S. Batzoglou, and A. Sidow. Quantitative estimates of sequence divergence for comparative analyses of mammalian genomes. *Genome Res.*, 13:813–820, May 2003.

[10] I. Dubchak, M. Brudno, G. G. Loots, L. Pachter, C. Mayor, E. M. Rubin, and K. A. Frazer. Active conservation of noncoding sequences revealed by three-way species comparisons. *Genome Res.*, 10:1304–1306, Sep 2000.

[11] L. Elnitski, V. X. Jin, P. J. Farnham, and S. J. Jones. Locating mammalian transcription factor binding sites: a survey of computational and experimental techniques. *Genome Res.*, 16:1455–1464, Dec 2006.

[12] G. B. Fogel, D. G. Weekes, G. Varga, E. R. Dow, H. B. Harlow, J. E. Onyia, and C. Su. Discovery of sequence motifs related to coexpression of genes using evolutionary computation. *Nucleic Acids Res.*, 32:3826–3835, 2004.

[13] M. C. Frith, Y. Fu, L. Yu, J. F. Chen, U. Hansen, and Z. Weng. Detection of functional DNA motifs via statistical over-representation. *Nucleic Acids Res.*, 32:1372–1381, 2004.

[14] S. J. Ho Sui, J. R. Mortimer, D. J. Arenillas, J. Brumm, C. J. Walsh, B. P. Kennedy, and W. W. Wasserman. oPOSSUM: identification of over-represented transcription factor binding sites in co-expressed genes. *Nucleic Acids Res.*, 33:3154–3164, 2005.

[15] M. A. Lones and A. M. Tyrrell. The evolutionary computation approach to motif discovery. In *in Biological Sequences,âĂİ Proc. Genetic and Evolutionary Computation Conf. (GECCO) Workshop Program, Workshop Biological Applications of Genetic and Evolutionary Computation*, pages 1–11, 2005.

[16] P. Monsieurs, G. Thijs, A. A. Fadda, S. C. De Keersmaecker, J. Vanderleyden, B. De Moor, and K. Marchal. More robust detection of motifs in coexpressed genes by using phylogenetic information. *BMC Bioinformatics*, 7:160, 2006.

[17] E. C. Rouchka and C. T. Hardin. rMotifGen: random motif generator for DNA and protein sequences. *BMC Bioinformatics*, 8:292, 2007.

[18] J. W. Thomas, J. W. Touchman, R. W. Blakesley, G. G. Bouffard, S. M. Beckstrom-Sternberg, E. H. Margulies, M. Blanchette, A. C. Siepel, P. J. Thomas, J. C. McDowell, B. Maskeri, N. F. Hansen, M. S. Schwartz, R. J. Weber, W. J. Kent, D. Karolchik, T. C. Bruen, R. Bevan, D. J. Cutler, S. Schwartz, L. Elnitski, J. R. Idol, A. B. Prasad, S. Q. Lee-Lin, V. V. Maduro, T. J. Summers, M. E. Portnoy, N. L. Dietrich, N. Akhter, K. Ayele, B. Benjamin, K. Cariaga, C. P. Brinkley, S. Y. Brooks, S. Granite, X. Guan, J. Gupta, P. Haghighi, S. L. Ho, M. C. Huang, E. Karlins, P. L. Laric, R. Legaspi, M. J. Lim, Q. L. Maduro, C. A. Masiello, S. D. Mastrian, J. C. McCloskey, R. Pearson, S. Stantripop, E. E. Tiongson, J. T. Tran, C. Tsurgeon, J. L. Vogt, M. A. Walker, K. D. Wetherby, L. S. Wiggins, A. C. Young, L. H. Zhang, K. Osoegawa, B. Zhu, B. Zhao, C. L. Shu, P. J. De Jong, C. E. Lawrence, A. F. Smit, A. Chakravarti, D. Haussler, P. Green, W. Miller, and E. D. Green. Comparative analyses of multi-species sequences from targeted genomic regions. *Nature*, 424:788–793, Aug 2003.

[19] J. A. Thompson and C. B. Congdon. GAMID: Using genetic algorithms for the inference of DNA motifs that are represented in only a subset of sequences of interest. In *Proceedings of the 2012 Congress on Evolutionary Computation. (CEC 2012)*, 2012. To be published.

[20] W. Thompson, E. C. Rouchka, and C. E. Lawrence. Gibbs Recursive Sampler: finding transcription factor binding sites. *Nucleic Acids Res.*, 31:3580–3585, Jul 2003.

[21] M. Tompa, N. Li, T. L. Bailey, G. M. Church, B. De Moor, E. Eskin, A. V. Favorov, M. C. Frith, Y. Fu, W. J. Kent, V. J. Makeev, A. A. Mironov, W. S. Noble, G. Pavesi, G. Pesole, M. Regnier, N. Simonis, S. Sinha, G. Thijs, J. van Helden, M. Vandenbogaert, Z. Weng, C. Workman, C. Ye, and Z. Zhu. Assessing computational tools for the discovery of transcription factor binding sites. *Nat. Biotechnol.*, 23:137–144, Jan 2005.

[22] T. Wang. Using PhyloCon to identify conserved regulatory motifs. *Curr Protoc Bioinformatics*, Chapter 2:Unit 2.12, Sep 2007.

[23] Z. Zhang and M. Gerstein. Of mice and men: phylogenetic footprinting aids the discovery of regulatory elements. *J. Biol.*, 2:11, 2003.

[24] J. Zheng, J. Wu, and Z. Sun. An approach to identify over-represented cis-elements in related sequences. *Nucleic Acids Res.*, 31:1995–2005, Apr 2003.