TFBS Identification by Position- and Consensus-led Genetic Algorithm with Local Filtering

Tak-Ming Chan, Kwong-Sak Leung, Kin-Hong Lee
Department of Computer Science & Engineering
The Chinese University of Hong Kong, Hong Kong
{tmchan, ksleung, khlee}@cse.cuhk.edu.hk

ABSTRACT

Identification of Transcription Factor Binding Site (TFBS) motifs in multiple DNA upstream sequences is important in understanding the mechanism of gene regulation. This identification problem is challenging because such motifs are usually weakly conserved due to evolutionary variation. Exhaustive search is intractable for finding long motifs because the combinatorial growth of the search space is exponential, thus heuristic methods are preferred. In this paper, we propose the Genetic Algorithm with Local Filtering (GALF) to address the problem, which combines and utilizes both position-led and consensus-led representations in present GA approaches. While position-led representation provides flexibility to move around the search space, it is likely to contain some "false positive" sites within an individual. This problem can be overcome by our local filtering operator, which employs consensus-led representation, while it needs less computation than alignments used in conventional consensus-led approaches. Thus both efficiency and accuracy can be achieved. The experimental results on real biological data show that our method can identify TFBSs more accurately and efficiently than other methods including GA-based ones, and is able to deal with relaxed motif widths with superior correctness.

Categories and Subject Descriptors

I.2.8 [**ARTIFICIAL INTELLIGENCE**]: Problem Solving, Control Methods, and Search—*Heuristic methods*; J.3 [**LIFE AND MEDICAL SCIENCES**]: Biology and genetics

General Terms

Algorithms, Performance

Keywords

Motif Discovery, TFBS Identification, Transcription Factors, Genetic Algorithm, Consensus, Local Search

GECCO'07, July 7–11, 2007, London, England, United Kingdom.
Copyright 2007 ACM 978-1-59593-697-4/07/0007 ...$5.00.

1. INTRODUCTION

Transcription Factor Binding Sites (TFBSs) are small nucleotide fragments in the promoter regions of genes within DNA sequences. TFBS is a crucial component in gene regulation, which affects the transcription process and finally the phenotype of organisms. Specifically, when certain protein called Transcription Factor binds the TFBS in the promoter region of the corresponding sequence, the transcription process is signaled and initiated. On the other hand, when other competing molecule interacts with the binding site so that the transcription factor fails to bind it, the transcription process will be inhibited. So it is important to identify those TFBSs in DNA sequences to decipher the mechanism of gene regulation.

So far the biological experiments such as DNA footprinting [5] and gel electrophoresis [6] have still been the most reliable and accurate methods, but they are especially expensive and very time-consuming. Alternatively, computational methods, namely *de novo* TFBS identification, have been proposed thanks to the availability of a great number of sequencing data. The fact is that a set of sequences carrying co-expressed genes, which are homologous genes expressed in different organisms, will have similar patterns of TFBSs, and these patterns can be generalized as a consensus, or a motif. And the more similar is a TFBS to the consensus, the stronger is the binding strength, which indicates a more significant promoter [8]. Thus with a collection of upstream sequences of the co-expressed genes (the sequence cut out before the genes. Notice this is a general case, for more sophisticated cases please refer to [23]), it is possible to discover TFBS motifs by comparing these sequences and extracting the similar small subsequences or fragments from each sequence.

However, the challenge lies in that subject to evolution and mutation, these motifs are weakly conserved, rendering simple string comparison methods helpless. On the other hand, exhaustive search of all combinations are not feasible due to the exponential growth of computation along with the increasing problem size. Thus constrained deterministic methods and heuristic methods are proposed. Deterministic methods with constraints model the problem as (l, d) motif discovery problem and employ approximate string matching algorithms (e.g. [2]). Here l is the length of the target motif and d is the maximal errors (usually substitutions in this problem) allowed between two extracted fragments. However, in real application, d is usually poorly defined. With a small d, weakly conserved motifs will be missing, while with a large d, the "over predict" problem is introduced

– a large amount of false positives will be produced. In addition, such methods often do not distinguish the quality between all outputs, and as a result further analysis is needed. Heuristic Multiple Sequence Alignment (MSA) methods also provide some insight into the problem because similar fragments are aligned together and they may be motifs. However, they do not generalize the aligned sequence data either, so they are also of limited help to the TFBS identification problem. Specifically for motif discovery, machine learning methods are proposed such as Expectation Maximization (e.g., MEME [1]), stochastic Gibbs sampling [15, 13] and Hidden Markov Models [21]. They show some particularly useful applications to locate the motifs we are interested in. The potential drawbacks of these methods include that they are sensitive to initial parameter setting, and that they may be trapped in local optima since many of them perform local search only [17]. Another caution is that with the employment of training data, they may bias those data because well annotated data are especially not sufficient for the current time, and thus may not reflect the complete biological truth.

Promisingly, evolutionary computation (EC) methods [4], especially genetic algorithms (GAs) [14, 3, 7, 18, 19, 22] are proposed to deal with TFBS identification problem. There are several advantages of GA compared with the conventional motif discovery methods [17]. First of all, GA carries out global search rather than local search, which is more likely to locate the global optima though it is not always guaranteed. Many conventional Bioinformatics methods (e.g. typical MSA, Expectation Maximization) perform local search and tend to converge to sub-optimal results, which often provide no biologically meaningful information. Secondly, as a general-purpose method, GA is also advantageous in the flexibility of representation and scoring. For example, fitness function can be based on probability distributions or on the similarity of substrings, which can get rid of the bias of training patterns. Thirdly, though GA is relatively slow compared with some heuristic methods, GA provides good scaling property when the problem size grows typically large. And this is appropriate for the real case where the lengths of sequences can vary from a hundred base pairs (bp) from prokaryotes to tens of thousands bp from eukaryotes.

The GA methods for TFBS identification can be categorized into two types by the different representations. The consensus-led approaches [19, 14, 18] are the ones in which the individual is encoded as the potential consensus, represented by A, T, C and G. On the other hand, the position-led approaches [3, 22] are those whose individual representation is an array storing the possible starting positions of TFBS in each sequence. The individuals of consensus-led methods can be randomly generated or picked up randomly from the subsequences with motif length among all the input sequences. One disadvantage of consensus-led approaches is that they require scanning all sequences to align each one to the consensus when evaluating a single individual, which imposes intensive computational load. Additionally, when the consensus happens to be a shifted version of the true one, they have no easy techniques to correct the consensus. Position-led approaches have more flexibility to move around the search space compared with consensus-led ones, because it is free to change any starting position, which consequently changes the motif configuration, and it is easy to shift the motif by changing all positions with a same small step. However, the representation cannot provide a global view of the quality of each TFBS position and cannot get rid of a small portion of the unsuitable positions easily.

In this paper, a new GA is proposed to complement the position-led and consensus-led representations, which not only makes use of their advantages but also tries to avoid their drawbacks. The proposed position- and consensus-led Genetic Algorithm with Local Filtering (GALF) maintains the flexibility of position representation and at the same time employs the consensus concept to guide local filtering for efficient refinement of the TFBS motifs. We evaluate the GALF and compare it with several methods including other GAs, and the experimental results show that our method is both accurate and efficient in the identification of real TFBSs.

The remainder of the paper is organized as follows. The following section gives details about the GALF approach. In Section 3, the experimental results will be reported. Section 4 discusses some issues of concern and we conclude this paper in Section 5.

2. THE PROPOSED APPROACH

In GALF, the basic representation of position-led type is maintained for its flexibility to explore the search space easily. The final fitness function is also calculated based on the position-led individuals. Meanwhile, the consensus with respect to an individual is also considered, and similarity scores are calculated and used for local filtering. Those "false positives" within an position-led individual in terms of similarity to the consensus will be filtered out, and the particular sequences will be scanned to choose a best replacement by local search. By doing this an individual can be refined efficiently.

2.1 Representation

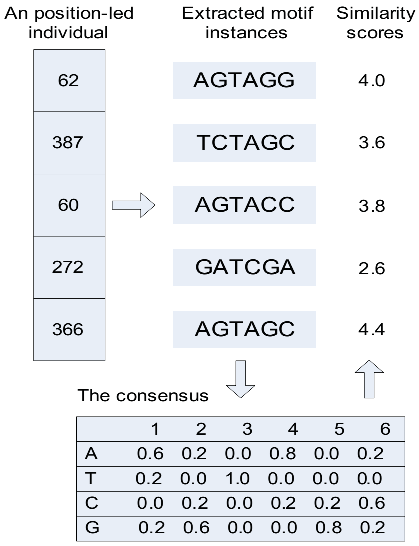
For an individual, the basic representation is the position-led one, which is an array storing the starting position in each sequence. We use integer instead of binary encoding. Starting from each position in an individual, a subsequence with the motif width can be extracted, and is called a motif instance. The consensus of an individual is represented by a Position-specific Weight Matrix (PWM) generated from the motif instances. Each cell in the PWM indicates the normalized frequency of the nucleotide in a particular position of the motif instances. An example of the two different representations is illustrated in Figure 1. For instance, the cell at the first row and the first column is the frequency of nucleotide A in motif position 1, which is $(1 + 0 + 1 + 0 + 1)/5 = 0.6$.

2.2 Fitness Evaluation

The fitness function we adopt for each individual is its information content [20]. It is widely employed (e.g. [3, 20]) or slightly revised (e.g. [18, 10]) to evaluate the potential motifs. For each position i in the extracted motif instances, the positional information content is

$$IC_i = \sum_b f_b \log \frac{f_b}{p_b} \tag{1}$$

where f_b is the observed frequency of nucleotide b on the column and p_b is the background frequency of the same nu-



	1	2	3	4	5	6
A	0.6	0.2	0.0	0.8	0.0	0.2
T	0.2	0.0	1.0	0.0	0.0	0.0
C	0.0	0.2	0.0	0.2	0.2	0.6
G	0.2	0.6	0.0	0.0	0.8	0.2

Figure 1: The position and consensus representations of an artificial individual and the similarities of its motif instances to the consensus

cleotide. The summation is taken over the four possible types of nucleotides ($b \in \{A, T, C, G\}$).

And the fitness is the sum of positional information content which has the following form:

$$fitness = \sum_{i=1}^{W} IC_i \qquad (2)$$

where W is the motif width.

Though known regulatory motifs do not always have highest information content at every base position [10], the sum of positional information content is till a good measure to reflect the overall conservations since for the moment no completely satisfactory measurement exists. And since we focus on the strongest motifs in each sequence, it is tolerant that the less conserved motifs with relatively lower information content are ignored for the moment and it is easy to rediscover them in the post-processing, which is considered in the discussion part.

As for the consensus representation, we use the similarity score to pick out those "false positives" of motif instances from a position-led individual. The instance similarity is calculated as the sum of the score of each corresponding letter in the PWM of the consensus:

$$similarity = \sum_{i=1}^{W} PWM(b_i, i) \qquad (3)$$

where b_i is the nucleotide in position i of the motif instance, and $PWM(b_i, i)$ is the score of b_i at position i in the PWM.

The artificial example is shown in the right part of Figure 1. For example, the score of the second motif instance $TCTAGC$ is calculated as $0.2{+}0.2{+}1.0{+}0.8{+}0.8{+}0.6 = 3.6$.

2.3 Selection and Genetic Operators for Reproduction

Binary tournament is employed for parent selection. In particular, when choosing a parent, we randomly pick up two individuals and choose the one with higher fitness. The purpose is to maintain appropriate selection pressure under which some of the currently unfit individuals have the chance to reproduce and may yield robust offspring in further generations.

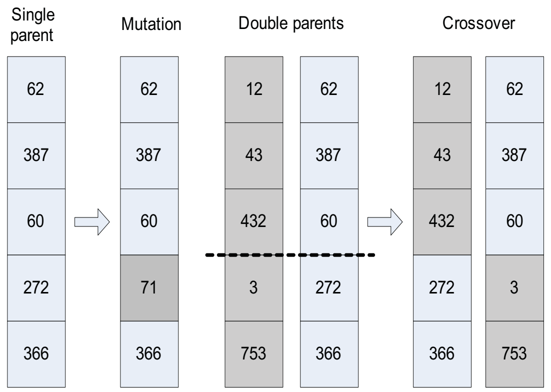


Figure 2: Mutation and crossover in reproduction

For reproduction, single-point mutation for a single parent and single-point crossover for double parents are applied. Mutation and crossover are performed with a totaling probability of 1. While mutation is chosen, one of the positions of the single parent will be shifted randomly. While crossover is applied, a crossover point is chosen at random from $[1, SeqNum - 1]$, where $SeqNum$ is the number of sequences. Then the segments of the two parents after the crossover point are swapped, yielding two children. One of them will be chosen at random as the offspring. The two genetic operators are illustrated in Figure 2. Multi-point mutation and crossover are also possible, but we just keep the single-point ones for we have local filtering to perform variations in the manner of directly improving the quality of an individual. After reproduction to generate offspring, the population is increased by a half for replacement.

2.4 Local Filtering Operator

One of the feature operators in GALF is the local filtering operator, which can filter out the "false positives" in a position-led individual in terms of the motif instances similarities to the consensus.

One dilemma of position-led GA approaches is that an individual may be made up of a portion of well located positions carrying highest similarities between the corresponding motif instances, but yet another portion is "false positives" which are poorly aligned to the consensus. Under such situation, genetic operators cannot efficiently escape from the local optimum because if any one or more members in the best portion are affected, the fitness may degrade significantly. And modifying those "false positives" purely by genetic operators can take generations. So it is desirable if the well identified portion can be distinguished from those "false positives". Consensus-led approaches achieve this by aligning all the sequences to the consensus in each generation [19, 14, 18], which imposes heavy computation. Yet it is challenging to have a criterion to determine the "false positives" in a position-led representation, though it provides more flexibility to vary than the consensus-led one.

In GALF, this problem is handled by the local filtering based on the consensus represented by PWM. Firstly, the motif instances within an individual is ranked by their similarity scores to the consensus. Secondly, the sequence containing the instance with the lowest similarity score is scanned. Among all possible starting sites of the instance, the one giving the best similarity to the consensus is chosen. If the rank does not change, which means this best instance is not better than its originally preceding instance from the other sequence in terms of similarity score, then the local filtering is stopped. Else the preceding instance is now the worst, and the sequence containing it is selected and scanned as is in the first step. This is iterated until the rank does not change or the sequence containing the originally second best instance is scanned. Notice that the PWM won't be updated before the local filtering is finished for two purposes. One is to save computational load compared with on-line update, and the other is to try not to be too greedy. On-line update may also be tried and tested. The pseudo-code is shown in Table 1.

Take Figure 1 as an example, after sorting the similarity scores, the instance from sequence 4 is the worst (2.6) and its preceding one (3.6) is from sequence 2. So sequence 4 is scanned for the best instance against the consensus. Suppose $AGTAGG$ (4.0) is found, since it is better than $TCTAGC$ (3.6) from sequence 2, sequence 2 is scanned until for some sequence, the best instance found is still worse than its preceding one. For example, if the best instance in sequence 2 is not better then its preceding instance $AGTACC$ (3.8) from sequence 3, local filtering is stopped.

This local search operator avoids the difficulty to determine the "false positives" by thresholds, but rather filter them by scanning the portions with worst similarity scores to the consensus iteratively. This technique does not require the full scan of all sequences in the consensus-led approaches. When an individual is subject to the evolutionary process, only a small number of "false positives" need to be filtered and only a few sequences need to be scanned. Since this operator is greedy to some degree, in order to keep the contribution of evolutionary process, it is only triggered once after certain generation intervals and applied to those newly generated or modified individuals which have never been filtered before.

2.5 Replacement Strategy

Replacement is applied to keep the population size to be constant after the increase of individuals during reproduction. Before replacement, all duplicate individuals will be removed to avoid too fast take-over. This is done by assigning an arbitrarily low fitness to those duplicates. The replacement strategy used here is K-tournament from [4]. Each individual competes with K randomly chosen other individuals, and scores a win if its fitness is higher than its competitor. K is user defined and we fix it 10 in our implementation. The number of wins of each individual are recorded and ranked, and when there is a tie in the number of wins between two different individuals, they are re-ranked by their fitness. Those whose final rankings are beyond the desired population size will be eliminated. Different from elitism directly on the fitness values, this scheme maintains some individuals which are not current elitists to fertilize with mutation or crossover in the future generations.

Table 1: Pseudo-code of local filtering operator

```
Input: Individual $P$
Notation: $P_i$ is the position of Sequence $i$ in $P$;
  $S(P_i)$ is the similarity score of the motif instance
  extracted from $P_i$; $Seq$ is the sequence number.
LOCAL_FILTER($P$)
{
  Sort the motif instances of $P$ by $S(\cdot)$ and obtain
  their sequence indices Ind(1), Ind(2), ... Ind(Seq)
  ($S(P_{Ind(1)})$ is the highest score and
  $S(P_{Ind(Seq)})$ is the lowest)
  for $k = Seq$ to 2, $k--$
  {
    Scan sequence $Ind(k)$ to get the best $S(P'_{Ind(k)})$;
    $P_{Ind(k)} = P'_{Ind(k)}$;
    if ($S(P_{Ind(k)}) <= S(P_{Ind(k-1)})$)
      Break;
  }
}
```

Table 2: Pseudo-code of shift operator

```
Input: Individual $P$, maximal possible shift $S$
SHIFT($P$,$S$)
{
  for $k = 1$ to $S$, $k++$
  {
    $P_{k+}$ = Shift $P$ by $+k$;
    $P_{k-}$ = Shift $P$ by $-k$;
    if (fitness($P_{k+}$)>fitness($P$) AND
      fitness($P_{k+}$)>=fitness($P_{k-}$))
      Return $P_{k+}$;
    if (fitness($P_{k-}$)>fitness($P$) AND
      fitness($P_{k-}$)>fitness($P_{k+}$))
      Return $P_{k-}$;
  }
  Return $P$;
}
```

2.6 Shift Operator

The "Phase Problem", stating the situation that the solution is a shifted version of the global optimum [13], is also under consideration. A shift operator is applied to deal with it. When the individual with best fitness stagnates, which means it does not change after certain generations (we set it as the generations of stagnation), a small number of shifts (all of the positions of the individual are moved in either direction by the same bases) are tested for improvement of fitness. Based on the gain of fitness, the smallest shift with positive gain will be chosen. If both directions of the same shift number achieve improvement, we choose the direction with better gain. The pseudo-code of shift operator is shown in Table 2. This moderate shift operator is to prevent a drastic shift which may drag the solutions to local optima too fast before convergence. While the fittest individual is really slightly shifted from the global optimum, shifts performed after several times of stagnation are enough to lead it to the correct location.

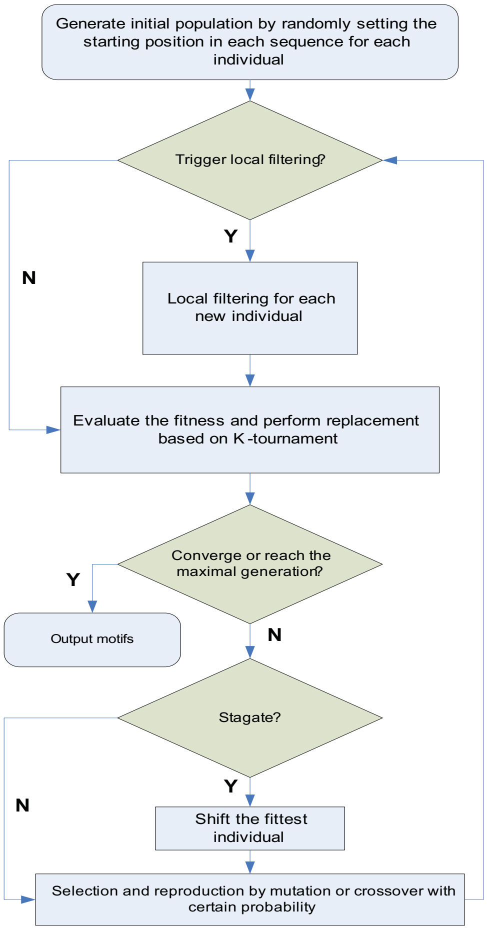

Generate initial population by randomly setting the starting position in each sequence for each individual
Trigger local filtering?
Y
N
Local filtering for each new individual
Evaluate the fitness and perform replacement based on K-tournament
Converge or reach the maximal generation?
Y
Output motifs
N
Stagate?
Y
N
Shift the fittest individual
Selection and reproduction by mutation or crossover with certain probability

Figure 3: The whole procedure of GALF

2.7 The Implementation

In our implementation, the population is initialized by randomly generating the starting positions of binding sites in each sequence and each set of positions representing an individual is stored in an array. The population size is set as 500 and the offspring size as 250. In the first generation, local filtering is performed to refine the population quality. The maximal number of generations is set to 200, and if the fittest individual remains the same for 50 generations it is thought to be converged. For the shift operator, generations of stagnation is set to be 10. This is also the generation interval to trigger local filtering. The whole procedure is shown in Figure 3.

3. EXPERIMENTAL RESULTS

In this section, we present the experiment results of 3 sub-sections of benchmark data tested on GALF and other methods including GAs.

3.1 CRP Binding Sites

First we test the performance of GALF on the dataset of cyclic-AMP receptor protein (CRP) binding sites, which consists of 18 sequences of 105 bps in length [20]. 23 binding sites are identified by DNA footprinting with motif width of 22. We test different mutation probability settings ranging from 0.1 to 0.9 with step 0.2.

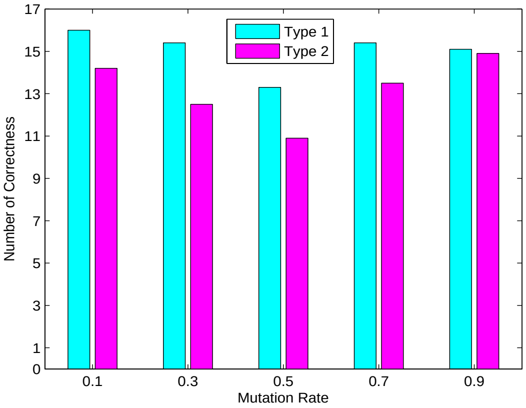


Figure 4: The average $Type1$ and $Type2$ correctness for different mutation probabilities, each setting run 10 times

Two types of correctness criteria are used in the evaluation. $Type1$ correctness is a more comprehensive and looser one allowing shifts [3, 22], which treats a binding site is correctly identified with tolerance of shifting up to 3 bp in either direction (in [3], shift of 5 bp is allowed). $Type2$ correctness is stricter, which only counts the exactly matched binding sites. Running GALF for 10 times for each setting, we find that with mutation probability 0.1 (thus crossover probability 0.9) the average $Type1$ correctness is highest, with a relatively high average $Type2$ correctness, which is shown in Figure 4. Since $Type1$ correctness is widely accepted, we fix the best mutation probability 0.1 in the following experiments. Our discovery is different from [3], which surprisingly favored a low crossover probability. One probable reason is that the local filtering in our method demonstrates the exploiting capacity and a high crossover probability contributes more to the exploration of the search space. Mutation maintains the variation which may also improve the individuals.

We also record the average number of scanned sequences per individual every time when local filtering is triggered of the 10 runs with the determined mutation probability 0.1, and the result is shown in Figure 5. The average number drops to a low level as generation increases, which is the evidence that evolution process leads the population towards a fitter one with less and less "false positives" in each individual.

It is noticeable that comparing motif discovery tools is difficult because the results are affected by the setting of parameters as well as the choice of data sets. In order not to favor our method, we perform the experiment on the same datasets used in other representative GA methods [3, 18, 22] and compare our results with theirs in which the parameters are considered well tuned according to their own expertise to give best performance and the experiments are well performed.

On this CRP dataset, GALF is compared to Gibbs Motif Sampler [15], BioProspector [16] and MDGA [3] and the results are illustrated in Table 3. The second column lists all the starting positions of the true sites and there may be more than one sites within one sequence. For each method, the

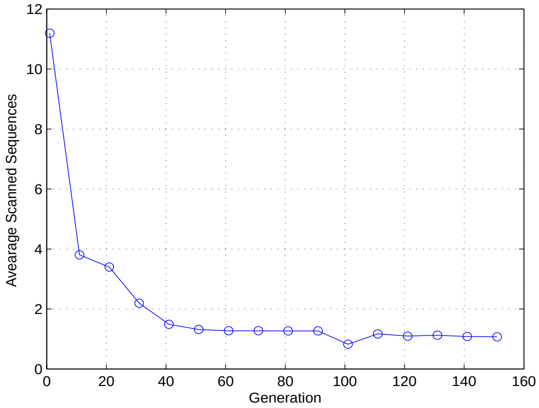


Figure 5: The average scanned sequences per individual in triggered generations for 10 runs

shifts of the predicted starting positions of sites compared to the real ones are listed in the table, where 0 means the predicted site is exactly the true site. In the case with multiple sites, the shift from the nearest site is recorded. The values in the parentheses indicate the predicted starting positions. While GALF and MDGA find the most of the motifs in terms of $Type1$ correctness (all those with shifts not more than 3), GALF locates the binding sites more accurately than MDGA with significantly fewer shifts. 17 correct sites of the 18 sequences are exactly located without any shifts in GALF. Only in one sequence GALF cannot locate the exact TFBS and the shift of error is only 4, which is significantly smaller compared to the -28 shift error in MDGA.

3.2 Relaxed Motif Width

In [18] (for convenience, we call the approach ConGA for it is consensus-led) another CRP motif dataset [18] with six sequences of 502 bp is experimented and the motifs embedded are 19 in width. The motifs embedded in these sequence and their starting positions (the beginning of a sequence is labelled 0) are:

$AATGTTATCCACATCACAA$	36
$AAAGTGAACCATATCTCAA$	64
$CTTGTGATTCAGATCACAA$	214
$TGTGTGATCGTCATCACAA$	59
$TGTGTGAAGTTGATCACAA$	37
$TTGGTGAGGAACTTAACAA$	314

ConGA is able to discover 3 of them and binary GA cannot identify any. GALF is tested on this data set and find that 5 binding sites out of 6 are identified with $Type1$ correctness (Table 4). And for a more general case, the motif width in real application may not be known exactly in prior. For example, in this experiment width 19 is used for CRP instead of 22 used in the previous experiment. So we relax the motif width to see if GALF can still identify the region of the true motifs. We set the motif width to vary from 16 to 23 at a step of 1. As Table 4 shows, within this range of motif variations, GALF is still able to identify $4-5$ of the true motifs, which demonstrates its capability to deal with relaxed motif widths in real applications. This potential can be further developed to be more formal in the future work.

Table 4: Number of CRP binding sites identified (in terms of $Type1$ correctness) in ConGA, binary GA (BGA) and GALF. W is the motif width and GALF is tested with different widths ranging from 19 to 23.

	W	1	2	3	4	5	6	$Type1$
True	19	36	64	214	59	37	314	
ConGA	19	136	64	375	59	37	137	3
BGA	19	4	0	264	16	11	69	0
GALF	19	38	66	216	61	39	139	**5**
	16	39	67	217	62	40	260	**5**
	17	39	67	217	62	40	260	**5**
	18	39	67	217	62	40	260	**5**
	20	38	66	216	61	39	139	**5**
	21	36	64	375	59	37	137	4
	22	35	63	374	58	36	136	4
	23	34	62	373	57	35	135	4

3.3 ERE and E2F motifs

Besides the CRP motifs, experimental results of GALF on two other datasets are also reported. The estrogen receptor element (ERE) dataset [12] contains 25 sequence of 200 bp, each of which is embedded with a single known binding site ERE which activates gene expression in response to estradiol. The E2F dataset [11] contains 25 mammalian sequences, 200 bp in length, in which 27 binding sites are included for transcription factors in the E2F family. The motif width is 13 for these two datasets. The measure is the F-score [22] combining precision and recall. For motif discovery problem, let N_c be the number of correctly predicted motif sites, N_p the number of all the predicted motif sites and N_t the number of true motif sites embedded in the sequences. Then precision and recall have the following forms:

$$Precision = \frac{N_c}{N_p} \tag{4}$$

$$Recall = \frac{N_c}{N_t} \tag{5}$$

And the F-score is

$$F = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6}$$

Again shifting up to 3 bp is allowed ($Type1$ correctness) for identifying a correct site. Comparisons are among GALF, GAME, MEME, BioProspector (BioPros.) and BioOptimizer [9], which is an optimization program which can be associated with MEME (BioOpt.M.) or BioProspector (BioOpt. B.). The results are reported in Table 5. GALF and GAME obtain the the best F-scores overall. GAME [22] is best in one case, but for all the experiments it requires 3000 generations to achieve the comparable results, whereas only 200 generations are needed in GALF due to the fast scanning and refinement by local filtering.

Table 3: Correctness in terms of shifts from the nearest starting positions of true CRP binding sites and the total $Type1$ correctness of Gibbs Motif Sampler (Gibbs.), BioProspector (BioPros.), MDGA and GALF. The predicted starting positions of sites are shown in the parentheses.

Seq No.	True Sites	Gibbs.	BioPros.	MDGA	GALF
1	17, 61	-2 (59)	2 (63)	1 (62)	**0 (61)**
2	17, 55	-2 (53)	2 (57)	1 (56)	**0 (55)**
3	76	-2 (74)	2 (78)	1 (77)	**0 (76)**
4	63	-4 (59)	2 (65)	1 (64)	**0 (63)**
5	50	-39 (11)	2 (52)	1 (51)	**0 (50)**
6	7, 60	-2 (5)	2 (9)	1 (8)	**0 (7)**
7	42	-2 (40)	-16 (26)	1 (43)	**0 (42)**
8	39	-2 (37)	2 (41)	1 (40)	**0 (39)**
9	9, 81	-2 (7)	2 (11)	1 (10)	**0 (9)**
10	14	-2 (12)	2 (16)	1 (15)	**0 (14)**
11	61	-2 (59)	2 (63)	1 (62)	**0 (61)**
12	41	6 (47)	2 (43)	1 (42)	**0 (41)**
13	48	-2 (46)	2 (50)	1 (49)	**0 (48)**
14	71	-2 (69)	2 (73)	1 (72)	**0 (71)**
15	17	-2 (15)	2 (19)	1 (18)	**0 (17)**
16	53	-4 (49)	2 (55)	1 (54)	**0 (53)**
17	1, 84	24 (25)	-16 (68)	-28 (56)	**4 (5)**
18	78	-4 (74)	2 (80)	1 (77)	**0 (78)**
Total $Type1$ Correctness		12	16	**17**	**17**

Table 5: Experimental results on ERE and E2F datasets

Data	Appro.	Precision	Recall	F-score
CRP	GALF	17/18	17/23	**0.83**
	GAME	16/17	16/23	0.80
	BioOpt.M.	12/13	12/23	0.67
	BioOpt.B.	12/13	12/23	0.67
	MEME	12/13	12/23	0.67
	BioPros.	16/18	16/23	0.78
ERE	GALF	21/25	21/25	**0.84**
	GAME	19/26	19/25	0.75
	BioOpt.M.	17/22	17/25	0.72
	BioOpt.B.	18/23	18/25	0.75
	MEME	15/17	15/25	0.71
	BioPros.	14/16	14/25	0.68
E2F	GALF	20/25	20/27	0.77
	GAME	23/24	23/27	**0.90**
	BioOpt.M.	20/27	20/27	0.74
	BioOpt.B.	19/27	19/27	0.70
	MEME	19/23	19/27	0.76
	BioPros.	11/21	11/27	0.46

4. DISCUSSION

It has been demonstrated that position-led GA approaches have the capacity to find out true motifs in a collection of sequences. Consensus-led GA approaches also achieve this by full sequence alignments which are computational intensive. However, the efficiency can be significantly improved by the combination of both representations and local filtering. Local filtering using consensus is able to refine a position-led individual to get rid of its "false positives" instantly, and at the same time only requires scans on a small number of the sequences. The capacity of local filtering is impressive and stable, and by employing a moderate shifting operator, the accuracy is further refined.

Multiple occurrences of weaker motifs within one sequence may be desired by some practitioners. Though it is believed that the most resembling motifs in the sequences are of most significance for they have the strongest binding strength [8], it is also easy to modify GALF to identify multiple TFBSs in one sequence. Performing sequence scan similar to the local filtering on the optimum found by GALF is sufficient to dig out those weaker motifs which have relatively high similarity to the consensus.

One assumption of some GAs such as MDGA and GALF is that each sequence is embedded with at least one motif. But it may not always be the case when some sequences carrying no motifs are collected. This can be overcome by developing a more sophisticated fitness function as well as the local filtering to handle zero motifs, or randomly generating null positions in case some sequences may not contain any motifs. Since GA is a generic framework, specific modifications are easy to implement in future work. More useful prior knowledge of the motifs themselves is also desirable, which may cope with the situation that when the motifs are very short, multiple patterns may exist with the same high fitness of information content, some of which may be false positives.

5. CONCLUSION

In this paper, a novel genetic algorithm GALF is proposed, which utilizes and complements the position- and consensus-led representations as well as introduces the local filtering which efficiently speeds up the search and improves the prediction accuracy. Experimental results show the superior performance of GALF compared with several other methods including GA-based ones. GALF also has the potential to accept a relaxed motif width. Further refinement of GALF includes post-processing to extract multiple weaker motifs, more sophisticated fitness function and filtering operator to deal with zero motifs within one sequence, and more precise criteria to distinguish and evaluate the redundant output if a very short pattern needs to be found.

6. ACKNOWLEDGMENTS

The authors are grateful to Topon Kumar Paul and Zhi Wei for their valuable comments and datasets. This research is partially supported by the grants from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. CUHK4173/04E and CUHK4132/05E).

7. REFERENCES

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

[2] P. Bieganski, J. Riedl, J. V. Carlis, and E. Retzel. Generalized suffix trees for biological sequence data: applications and implementations. In *Proc. of the 27th Hawaii Int. Conf. on Systems Sci.*, pages 35–44, 1994.

[3] D. Che, Y. Song, and K. Rasheed. MDGA: motif discovery using a genetic algorithm. In *GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation*, pages 447–452, 2005.

[4] 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(13):3826–3835, 2004.

[5] D. J. Galas and A. Schmitz. DNAse footprinting: a simple method for the detection of protein-DNA binding specificity. *Nucleic Acids Res.*, 5(9):3157–3170, September 1987.

[6] M. M. Garner and A. Revzin. A gel electrophoresis method for quantifying the binding of proteins to specific DNA regions: application to components of the escherichia coli lactose operon regulatory system. *Nucleic Acids Res.*, 9(13):3047–3060, July 1981.

[7] J. Gertz, L. Riles, P. Turnbaugh, S. W. Ho, and B. A. Cohen. Discovery, validation, and genetic dissection of transcription factor binding sites by comparative and functional genomics. *Genome Research*, 15:1145–1152, 2005.

[8] D. L. Hartl and E. W. Jones. *Genetics, Analysis of Genes and Genomes.* Jones and Bartlett Publishers, 6 edition, 2005.

[9] S. T. Jensen and J. S. Liu. BioOptimizer: a Bayesian scoring function approach to motif discovery. *Bioinformatics*, 20:1557–1564, 2004.

[10] K. J. Kechris, E. van Zwet, P. J. Bickel, and M. B. Eisen. Detecting DNA regulatory motifs by incorporating positional trends in information content. *Genome Biology*, 5(7):R50, June 2004.

[11] A. E. Kel, O. V. Kel-Margoulis, P. J. Farnham, S. M. Bartley, E. Wingender, and M. Q. Zhang. Computer-assisted identification of cell cycle-related genes: new targets for E2F transcription factors. *J. Mol. Biol.*, 309(1):99–120, 2001.

[12] C. M. Klinge. Estrogen receptor interaction with estrogen response elements. *Nucleic Acids Res.*, 29:2905–2919, 2001.

[13] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wooton. Detecting subtle sequence signals: a Gibbs sampling strategy for multiple alignment. *Science*, 262(8):208–214, October 1993.

[14] F. F. M. Liu, J. J. P. Tsai, R. M. Chen, S. N. Chen, and S. H. Shih. FMGA: Finding motifs by genetic algorithm. In *BIBE '04: Proceedings of the 4th IEEE Symposium on Bioinformatics and Bioengineering*, pages 459–466, 2004.

[15] J. S. Liu, A. F. Neuwald, and C. E. Lawrence. Bayesian models for multiple local sequence alignment and Gibbs sampling strategies. *J. Am. Stat. Assoc.*, 90(432):1156–1170, November 1995.

[16] X. Liu, D. L. Brutlag, and J. S. Liu. BioProspector: discovering conserved DNA motifs in upstream regulatory regions of co-expressed genes. In *Pac. Symp. Biocomput.*, volume 6, pages 127–138, 2001.

[17] M. A. Lones and A. M. Tyrrell. The evolutionary computation approach to motif discovery in biological sequences. In *GECCO '05: Proceedings of the 2005 workshops on Genetic and evolutionary computation*, pages 1–11, 2005.

[18] T. K. Paul and H. Iba. Identification of weak motifs in multiple biological sequences using genetic algorithm. In *GECCO '06: Proceedings of the 8th annual conference on Genetic and evolutionary computation*, pages 271–278, 2006.

[19] M. Stine, D. Dasgupta, and S. Mukatira. Motif discovery in upstream sequences of coordinately expressed genes. In *CEC '03: Evolutionary Computation, The 2003 Congress on*, volume 3, pages 1596–1603, 2003.

[20] G. D. Stormo. Computer methods for analyzing sequence recognition of nucleic acids. *Annu. Rev. BioChem.*, 17:241–263, 1988.

[21] G. Thijs, M. Lescot, K. Marchal, S. Rombauts, B. D. Moor, P. Rouze, and Y. Moreau. A higher-order background model improves the detection of promoter regulatory elements by Gibbs sampling. *Bioinformatics*, 17:1113–1122, 2001.

[22] Z. Wei and S. T. Jensen. GAME: detecting cis-regulatory elements using a genetic algorithm. *Bioinformatics*, 22(13):1577–1584, 2006.

[23] G. A. Wray, M. W. Hahn, E. Abouheif, J. P. Balhoff, M. Pizer, M. V. Rockman, and L. A. Romano. The evolution of transcriptional regulation in eukaryotes. *Mol. Biol. Evol.*, 20(9):1377–1419, 2003.