

GRAPHICAL REPRESENTATION OF BIOLOGICAL INFORMATION

HUANG ENLI

NATIONAL UNIVERSITY OF SINGAPORE

2005

**GRAPHICAL REPRESENTATION OF BIOLOGICAL
INFORMATION**

HUANG ENLI

(B.Eng.(Hons.), Nanyang Technological University)

**A THESIS SUBMITTED
FOR THE DEGREE OF MASTER OF ENGINEERING
DEPARTMENT OF MECHANICAL ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE**

2005

Acknowledgements

With a deep sense of gratitude, I wish to express my sincere thanks to my supervisor, Professor Vladimir B. Bajic, for his immense help in planning and executing the works in time. His company and assurance at the time of crisis would be remembered lifelong. Gratitude also goes to my co-supervisor Associate Professor Toh Siew Lok and Professor Nhan Phan Thien. Their valuable suggestions as final words during the course of work are greatly acknowledged.

My sincere thanks are given to Dr. Tang Shuisheng for various suggestions and also for help and encouragement during the research work. I specially thank Ms Zhang Guanglan, Ms Judice Koh, Mr Tan Sinlam, Dr Bijayalaxmi Mohanty, Vidhu Choudhary for the help extended to me when I approached them and the valuable discussion that I had with them during the course of research

I wish I would never forget the company I had from my fellow research scholars of Institutes of Informcom Research (I2R). In particular, I am thankful to Yang Liang, Manisha Brahmachary, Rajesh Chowdhary, Zuo Li for their help.

Finally, I acknowledge all persons in the Department of Mechanical Engineering at the National University of Singapore, for their efforts during my educating and I also extend my thanks to the staffs in I2R for their cooperating throughout the course of this research.

Table of Contents

Acknowledgements.....	I
List of Publications	VI
List of Figures	VII
List of Tables	X
List of Acronyms	XI
1.1 Background.....	1
1.2 Research goals and assumptions.....	7
1.3 Layout of the thesis.....	9
Chapter 2 Literature Review.....	11
2.1 Basic of Molecular Biology.....	12
2.1.1 DNA structure.....	13
2.1.2 Gene	14
2.1.3 Regulatory factors.....	15
2.1.4 TF binding sites.....	16
2.1.5 Promoter Fundamentals	16
2.1.6 Gene expression and transcription mechanism.....	18
2.2 Bioinformatics.....	21
2.2.1 Motif Prediction	21
2.2.2 Graphical presentations of various biological information.....	24
2.2.3 Graph drawing packages and applications.....	29
Chapter 3 Ab-initio Motif Discovery.....	34
3.1 A broader context of motif discovery: Gene Finding	34

3.2 Heuristic Algorithms in Motif Discovery	36
3.2.1 Expectation Maximization (EM) Algorithm.....	37
3.2.2 Genetic Algorithm (GA).....	44
3.3 Overall program flow-chart	55
Chapter 4 Transcription Start Site Viewer (TSSViewer)	57
4.1 Problem Statement	57
4.2 Objectives	58
4.3 System Description	59
4.4 Software Description	60
4.5 File Format.....	60
4.5 Program Flow.....	64
4.6 Comment on TSSViewer	66
Chapter 5 MotifBuilder and the web application.....	67
5.1 Problem Description	67
5.2 Objectives	68
5.3 MotifBuilder Description.....	68
5.4 Motif Report.....	69
5.5 Visual Presentation of Motif Information.....	71
5.5 Visual Presentation of Motifs	75
5.6 Web-based Application.....	77
5.6.1 Dragon Motif Search Tool.....	78
5.6.2 Procedures and Operations of Dragon Motif Search Tool.....	78
5.7 Other Applications	80

Chapter 6 TFMapper.....	83
6.1 Objectives of the Development.....	83
6.2 Software Description	84
6.3 Working Principle.....	86
6.4 Using TFMapper software	87
6.5 Input / Output File Information.....	88
6.6 Program Flow chart.....	91
6.7 Applications of TFMapper.....	92
Chapter 7 Discussions and Comments.....	98
7.1 Heuristic System Performance.....	98
7.1.1 Efficiency	98
7.1.2 Precision.....	101
7.2 Comments on graphical representation.....	102
Chapter 8 Conclusion and Further work.....	104
References.....	108
Appendix 1:.....	116

Summary

Biological information is complex due to numerous ways how biological entities affect each other. Human comprehension of this information is easier if the information is in a graphic form. However, different biological problems require different types of information to be presented and thus graphical information is dependent on the type of problem in question and equally on the type of data from which the representation is generated. In this study I focused on preparation of data for graphical representation and graphical presentation of information for several transcription regulation problems. The problems investigated were: a/ annotation of human promoters by transcription factor binding sites (TFBSs), b/ distribution of DNA motifs in a set of sequences, c/ networks of genes and associated TFBSs or motifs. In this process, a database of annotated human promoters with interactive graphical representation of the promoter content is developed where user can visualize distribution of individual TFBSs and pairs of TFBSs across the promoter and also find basic information on the TFBSs. Two novel heuristic models (based on expectation maximization and genetic algorithm) to identify motifs by ab-initio approach were developed and implemented. This allowed for the visualization of the distribution of motifs found across set of sequences and within individual sequences. Moreover, this served as a basis for producing data from which graphical representation of transcriptional regulatory networks were derived. The results developed in this study have been proven useful for the analysis of several transcription regulation problems as they allowed for inspection of complex relation between TFBSs/motifs and promoters/genes through relatively simple graphical representation.

List of Publications

VB Bajic, E Huang, L Yang, Modeling methodology for detection of regulatory motifs in DNA/RNA and proteins, *Int.J.Comp.Syst.Signals*, (accepted) 2005

L Yang, E Huang, VB Bajic, Some implementation issues of heuristic methods for motif extraction from DNA sequences, *Int.J.Comp.Syst.Signals*, 5(2) (in print) (2004)

E Huang, L Yang, R Chowdhary, A Kassim, VB Bajic, An algorithm for ab initio DNA motif detection, Chapter 4 in *Information Processing and Living Systems*, World Scientific, 611-614, 2005

Krishnan SPT, E Huang, L Yang, V B Bajic, Statistical Properties of region around PolyA sites in Human, 5th HUGO Pacific meeting and 6th Asia Pacific meeting on Human genetics, 17-20 November 2004, Singapore.

List of Figures

Figure 1.1 Illustration of the potential association of genes A and C via interconnecting gene B	3
Figure 1.2 Illustration of the associations between the genes through TFs whose binding site are found in the genes' promoters. The oval nodes represent TFs, while octagonal nodes represent target genes. The case corresponds to the mouse data.	5
Figure 2.1 Presentation of a double helix structure and chemical compound representation	13
Figure 2.2 Features of nucleotide: phosphate, pentose and base	14
Figure 2.3 General organization of the DNA sequence. Only the exons encode a functional peptide or RNA. The coding region accounts for about 3% of the total DNA in a human cell	15
Figure 2.4 A typical structure of promoter showing binding sites and promoter modules	17
Figure 2.5 Process of Eukaryotic Gene expressions	18
Figure 2.6 Assembly of the activator/promoter complex on the proximal and core promoter region. a) Schematic representation of the proximal promoter with these specific TF binding sites and the core promoter represented by the TATA box (black triangle) and the initiator region (INR). The transcription start site (TSS) is indicated by the angled arrow. b) Binding of the TFs and the TFIID complex (including the TAA box binding protein TBP). TBP binding induces a 90° bend in the promoter DNA. c) Subsequently the polymerase II/GTF complex is loaded to yield the complete initiation complex.....	20

Figure 2.7 Matrix based TF profile.....	25
Figure 2.8 Association of e different terms defined in PubMed documents. Documents were collected based on query “antimicrobial toll”. Antimicrobial peptides are important component of innate immune system in vertebrates. Gene with produce them are mainly controlled through the toll-like receptor pathway of which NF-kappaB is one of the key regulators. Text-mined information conveniently presents such associations.....	26
Figure 2.9 The snapshot of the CellDesigner 3.0	27
Figure 2.10 Snapshot of the multicontigview expression in ENSEMBL	28
Figure 3.1 Features of two point crossover.....	47
Figure 3.2 Features of one point mutation	47
Figure 3.3 Main program flow-chart	55
Figure 4.1 Snapshots of the TFBSs description entry.....	61
Figure 4.2 Snapshots of the output file	62
Figure 4.3 The content of pop up windows	64
Figure 4.4 TSSViewer Program Flow Chart.....	65
Figure5.1 Motif report from the heuristically search.....	69
Figure5.2 Tabular representation of the PWM for a motif family in the html file	70
Figure 5.3 Starting position distribution list for one group of motifs.....	71
Figure 5.4 HTML expression format for the position distribution chart	72
Figure 5.5 Motif distribution in the promoter region [-250,-1] relative to TSS, for mouse H4 histone gene group.	73
Figure 5.6 Interconnection Network between the motifs and sequences.....	75

Figure 5.7 Schematic presentation of the module for generation of reports that contain graphics	76
Figure 5.8 Snapshot of the Dragon motif search tool	77
Figure 5.9. Snapshot of the promoter content of ATF3 ortholog genes. In the case of human and rat (5.9c) there are more common promoter elements that have preserved positional organization, than is the case when human, mouse and rat are considered (5.9b). This suggests mouse specific solution in promoter composition for the ATF3 gene.....	81
Figure 6.1 Graphical user interface of the TFMapper	86
Figure 6.2 Translated Input file for Graphviz.....	88
Figure 6.3 Relation network for genes and TFBSs.....	89
Figure 6.4 Program flow chart for TFMapper	91
Figure 6.5. A subnetwork of interconnected genes from group of 17 very highly expressed in epithelial ovarian cancers. The link between the genes is made only if they share at least five PEs in their promoters.....	94
Figure 6.6. The network of genes that are highly expressed in epithelial ovarian cancer shown with PEs that potentially control these genes. The network is generated by TFMapper using four PEs (TCF11(+), AREB6(-), XPF-1(-), Kr(+)) as seed PEs...	95
Figure 6.7. A larger gene network that contain a subnetwork of genes associated with matrix metallopeptidase group.....	96
Figure 7.1 Report for motifs obtained.....	100

List of Tables

Table 3.1: Align pattern extracted from sequences	39
Table 3.2: PWM of the align motifs	39
Table 3.3: Normalized PWM.....	40
Table 3.4: Normalized PWM.....	52
Table 7.1 Search criteria for EM and GA for comparison.....	99
Table 7.2 Search criteria for MEME, EM and GA for comparison.....	102

List of Acronyms

TF: transcription factors

TFBSs: transcription factor binding sites

PE: promoter element

bp: base pair

nt: nucleotide

TAF: transcription accessory factors

GTF: general transcription factor

TIC: transcription initiation complex

TFIID: transcription factor IID

GA: genetic algorithm

EM: expectation maximization

IC: information content

PWM: position weight matrix

HMM: Hidden Markov Model

ORI: over-representation index

nt: nucleotide

DMB: Dragon Motif Builder

DMST: Dragon Motif Search Tool

TSS: Transcription Start Sites

Chapter 1 Introduction

1.1 Background

The last decade has witnessed the dawn of a new era of 'silicon-based' biology. It is the first time that it became possible to investigate and make comparative analyses of complete genomes. In its broadest sense, genome analysis is underpinned by a number of pivotal concepts concerning structural properties of DNA and RNA, regulatory elements, transcription, RNA processing, translation, processes of evolution, mechanism of protein folding and, crucially, the manifestation of protein function.

Currently, the completion of the Human Genome Project has generated huge amounts of genomic data. Additionally, other sequencing projects of other model organisms have also produced vast quantity of biological information. However, most of the genome data are ambiguous and uncharacterized, which become the major obstacle and challenge for the studies in molecular biology.

Biological processes themselves are very complex and involve interaction of numerous entities. For example, a gene can be activated only after specific biochemical conditions are provided in the cell. These involve numerous transcription factors (TFs) and the polymerase complex. Transcription initiation of gene A will require several TFs to interact with the promoter of gene A. Thus, genes that produce these TFs have to be active earlier, as their final protein products, TFs, are required in this transcription initiation process, and so on. This is just a short snapshot of a very simplistic description of just one of the fundamental processes in cell biology, the gene transcription initiation. As can be seen, even in this simplistic explanation, many components are involved. To be able to analyze such complex information and some aspects of their mutual relationships,

it is convenient to present information graphically in some suitable form. Unfortunately, this is not an easy task and, moreover, the convenience of such graphic presentation is problem specific.

Currently, great effort has been invested into suitable graphical representation of relevant information in bioinformatics, so as to cater for the various needs in biology research. Examples are system for the pathway processes for biological networks [38], structural gene and protein modeling [39, 40, 41], TF association information [14], etc. These systems utilize different graphical techniques and software to visualize the data and information.

In the field of molecular biology, the current research drive is towards understanding of relationships between different participants in various biochemical processes [1]. One of the most interesting, but equally one of the most complex and yet insufficiently understood processes is transcription regulation. It is great challenge for a biologist to comprehend to the full extent the complexity of these processes and to be able to identify the major players that are involved in the process. In the last two decades a lot of research has focused on the identification of the regulatory regions, promoters, enhancers, silencers, of various genes in many species [2]. Identification of gene promoters has been recognized as an important practical and research problem and a necessary step in understanding the underlying genetic regulatory mechanism. It also complements new gene discovery, as well as the reconstruction of transcriptional regulatory networks for genes of interest.

Ambiguities and lack of information in the text-format of biological data are a major problem for biologists to infer correct interpretation. Therefore, nowadays, there has been a significant progress in visual representation of biological data. The great interest in this field is because it does provide the human-readable diagrammatic visualization of relations between the examined entities, which is more convenient and more suitable for human interpretation. To illustrate the point, let us consider situation in Fig 1.1 that depicts association between genes A, B and C. Genes A and C are not directly associated, but both of them are associated to gene B. Consequently, one can hypothesize that A and C are associated indirectly via gene B. Moreover, one can hypothesize that gene B plays an important role in the link between A and C. One typical situation for this would be if B represents a TF that controls both gene A and gene C.

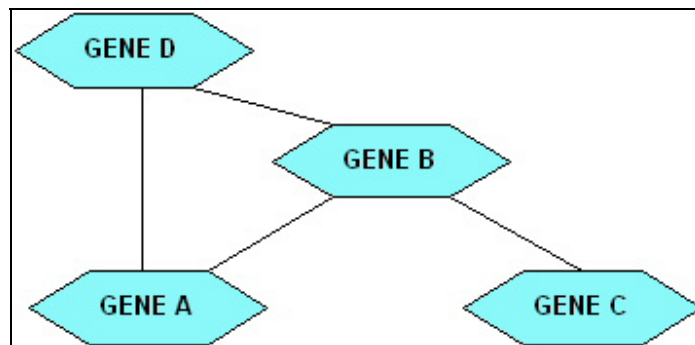


Figure 1.1 Illustration of the potential association of genes A and C via interconnecting gene B.

Although situation in Fig 1.1 is relatively simple, it demonstrates that suitable representation of data can enhance our ability to analyze that data and infer interesting possible relationships, which can further be subjected to more detailed analysis. One issue more that is worth mentioning in the context of graphical presentation of data is that

graphical presentation enhances our ability to observe complex structures in the relations contained in data.

This study focuses on graphical presentation of information related to transcription regulation. Analysis of genes regulatory regions involves both wet-lab experiments and frequently computational analyses. Wet-lab experiments are unfortunately laborious and expensive, and they are not that efficient and effective at large-scale for tasks of identification of genes regulatory regions in the uncharacterized genomic sequence. Thus, computational methods can substantially support and accelerate this process. One of the key problems in characterizing regulatory regions is identification of TF binding sites (TFBSs). These are short DNA segments that bind TF regulatory proteins. We can computationally predict many TFBSs with the aim to generate shortlist candidates for experimental verification. However, not every predicted TFBS will be subjected to experimental verification as there will be tens of thousands predicted across a large genome. Thus, one has to shortlist the most interesting ones. One way to evaluate which of the many predictions represent the interesting ones, is to try to analyze what is the collection of genes that contain such TFBSs and do they have something else in common. It is also possible of cross-check the list of such genes with results from specific microarray studies to see if such genes show similar behavior. But, while it is easy to say ‘try to analyze’ the set of genes, it does not reveal the way how to do this. One approach that can help a lot is to try to present the relations that such genes have between themselves with links through TFs whose binding sites are found in genes’ promoters, such as, for example in Fig. 1.2

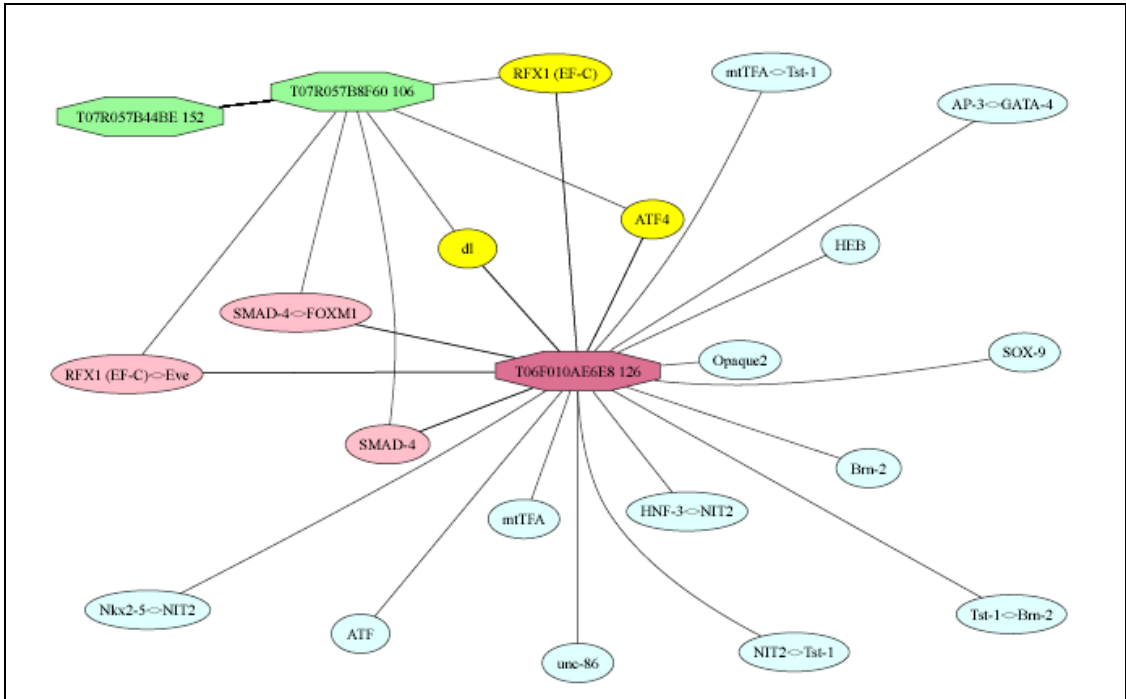


Figure 1.2 Illustration of the associations between the genes through TFs whose binding site are found in the genes' promoters. The oval nodes represent TFs, while octagonal nodes represent target genes. The case corresponds to the mouse data.

We also observe one other important problem. If one is interested to consider two genes associated with each other if their protein products are sufficiently similar (homology) then such two genes would be presented as two linked nodes, so very simple graphical representation. Such graphical representation would reflect association through gene product similarity and simple graphical representation will suffice. However, if one is interested in analyzing the transcriptional molecular mechanism that can provide such a link between these genes, then far more complex graphical representation results. Then, in addition to the genes of interest, we will also see TFs that potentially control them. Moreover, the data to be used for such graphical presentation has to contain such information. For example, even the best graphical representation software will not be able

to show links of genes and TFs that control them if the input data does not contain such information. Thus, we can conclude that graphical presentation is topic-specific (problem-specific), as it is suited to the goals of the analysis and it is also intimately intertwined with the data we provide for such representation.

The focus of this study is graphical presentation of predicted promoter elements (PEs) in promoter regions. PEs represent motifs and TFBSs found in the DNA sequence, as well as the DNA strand where they are found. So, a PE that relates to NF-kappaB TF that binds on the + DNA strand of one promoter could thus be denoted as NF-kappaB/+1. Of course, we can add more information such as the exact DNA location of the NF-kappaB motif. Promoter function is the result of the simultaneous effect of many composite functional modules involving numerous and specific combinations of PEs and their interaction with the available TFs, that is, two similar structural promoters might have different functional behaviors in terms of expression patterns of their respective genes depending on the internal organization of their PEs within their promoters. So, analysis of promoters is not a simple computational sequence-matching problem, because it not only involves the identification of potential PEs among the sequences, but also relies on the correlations of PEs among different promoters and consequently different genes. This, on the other hand, brings us directly to the utility of the graphical presentation of the part of that information, since tabular/text-type of information presentation will not be easy for interpretation. For example, if one wants to present interaction relations of the type as given in Fig 1.2 in the tabular/text format, it will be cumbersome and almost impossible to infer many connections between genes and numerous potential associations of TFs and genes in case there are a great number of

genes involved. Graphical representation, on the other hand, simplifies such insights in many cases.

Before one can graphically present information about promoters and the potential regulatory networks they determine, PEs have to be identified within promoter sequences. There are a number of computational techniques that have been proposed in the past to identify such elements. The techniques generally can be divided into two categories, namely, general PE and specific PE identification. The different aims for the identification methodologies will result in different predictions relative to the specificity and sensitivity of the identification method that is applied.

To extract the PEs effectively, local and global alignment techniques are developed mathematically to resolve the motif-mining problems. Different applications have been implemented in the BLAST [\[43\]](#), ClustalW [\[28\]](#), etc. These applications have become effective and reliable tools for the biologists to understand and analyze the biological information among sequences. However, these methods are not efficient for identification of short DNA sequences, such as TFs. For that reason many other specialized methods were developed. One set of such techniques deals with identification of short motifs from a set of DNA sequences [\[46\]](#). The other group of techniques uses mapping of TFBSs for which models exist in the form of position weight matrices [\[45\]](#).

1.2 Research goals and assumptions

This study aims at developing the suitable way to present graphically information related to PEs and more broadly transcription regulation, and associated with these the

methods to generate suitable data that can enable such graphical presentation. Therefore, the objectives of this research are to develop systems with the following functionalities:

- a) to perform the effective and efficient PE mining based on a heuristic algorithm
- b) to develop a suitable graphic representation of the basic PE/promoter information
- c) to develop graphical representation of networks for PEs identified in promoters.

The research project could be decomposed into two main research problems, each of which consists of several sub-problems as following:

1) Detecting the homogenous motifs among the sequences that include:

- a/ developing heuristic algorithms (expectation maximization and genetic algorithm) to extract the motifs;
- b/ applying hidden Markov model (HMM) to generate the background sequences;
- c/ determining the optimal motif prediction based on a statistical model.

2) Developing graphical applications for specific biological information presentation to:

- a/ convert the text format of a biological database related to promoter annotation into format that allows for direct graphic representation;
- b/ generate the graphic report for PEs, associated with the heuristic algorithm for motif detection;
- c/ construct some types of biological interaction networks related to transcription regulation problems, such as networks of genes linked through common PEs found in their promoters.

There are several main contributions of this research:

- 1) A database of annotated promoters with graphical presentation of the promoter content for a subset of human promoters is developed.
- 2) Two new efficient algorithms for determination of motif by ab-initio approach were developed; this served as a basis for generation of transcriptional regulatory networks.
- 3) A system for generating graphical presentation of transcriptional regulatory networks that can use motifs determined by ab initio methods or mapped TFBSs, is developed.

For the problem of identification of motifs by the ab initio approach we need to introduce the following assumptions related to the prediction process and promoter functionality:

1. A TF binds to a family of mutually very similar binding DNA sequences (these sequences we denote as homogeneous binding sequence set).
2. Heuristics is a suitable methodology for identifying PEs.
3. Promoters with similar structures contain many of the same PEs and their combinations.

Based on these assumptions, the heuristic algorithms were developed.

1.3 Layout of the thesis

The first chapter gives an introduction to the problem and explains the background and the research goals.

The second chapter is an overview of molecular biology topics of interest to problems in this study, especially those for the functionalities of promoters. Also, that chapter describes current bioinformatics/computational methodologies for promoter analysis.

In the third chapter, novel heuristic methods used to extract motifs from DNA sequences are described. Extensive research has been carried out to optimize the heuristic algorithms with control parameters. This is followed by the other developments and discussion of optimization for Hidden Markov Model and other statistical measures used. Based on the developed theoretical model, the computation results are presented to show the effects of different parameters and used to optimize the system to extract good motifs from the data.

In the fourth, fifth, and sixth chapters, the development of graphical applications is presented. Different graphical presentation techniques to describe the relationship of the TF and genes have been analyzed. Chapter 4 describes the diagrammatic graphical database presentation. Some web-base applications for the graphical reports associated with the heuristic algorithms are presented in Chapter 5. An interaction PEs-promoter network is explained and illustrated in Chapter 6.

The seventh chapter discusses the result of the developed motif search heuristic algorithms in terms of accuracy and efficiency, as well as comparison with some other methods. Moreover, we also comment the graphical presentation applications and techniques that we developed.

Finally, the last chapter presents general conclusions of this study, followed by a possible future work.

Chapter 2 Literature Review

As mentioned in Chapter 1, our interest is to explore the suitable graphical presentation techniques to visualize the complex biological information, especially in the topic of transcription regulation. Thus the fundamental knowledge on molecular biology and graphical tools are essential to assist us to develop the effective applications to cater for the current problems. In this chapter, we have literaturally reviewed on the basis of molecular biology, especially the transcription process. The computation algorithms to prepare motifs have been studied. Moreover, we also discussed the current graphical packages and their applications in the bioinformatics.

The field of molecular biology is related to macromolecules and macromolecular mechanisms that are found in living organisms. Examples of such mechanisms could be the molecular nature of gene including gene replication, mutation, and expression. The field of molecular biology is synthesis of many other fields including genetics, physics, chemistry, medicine, etc. that were focused on the problems of the structure and function of genes [3].

Several key discoveries have denoted various phases of molecular biology:

- Cellular basis of heredity (chromosomes).
- Molecular basis of heredity (DNA double helix).
- Informational basis of heredity (mechanism of decoding information contained in genes and discovery of recombinant DNA).

- Finally, genome sequencing and large-scale throughput technologies that enable insights into gene identification and gene structure.

After sequencing of the human genome that is completed in 2003, the current task is to understand and analyze that human genome sequence. This is complex and long-term task. In line of this, in our research, the correlation of TFs and genes were investigated by means of heuristic and statistic approaches and convenient graphical presentations of these results have been developed. The graphical format for result presentation makes the analysis of these results, inference of new information, and inference of relations between the involved entities, more convenient than non-graphical format or reports.

2.1 Basic of Molecular Biology

Cells are the basic units of living organisms, with the exception of viruses whose structure and function are different from cells. All cells are divided into two types: prokaryotic cells and eukaryotic cells.

The eukaryotic cell contains organelles, which are defined as membrane-bound structures such as nucleus, mitochondria, chloroplasts, endoplasmic reticulum (ER), Golgi apparatus, lysosomes, vacuoles, peroxisomes, etc. Prokaryotic cells do not have organelles. Eukaryotes are the organisms made up of eukaryotic cells. They include protista, fungi, animals and plants. Prokaryotes include archaeobacteria and eubacteria, which are single-cell organisms.

The genome is the complete set of genetic information inherited from the parents and comprises all the genes. The genome is physically presented in term of DNA, in which genes play a vital role by acting as a blueprint for the production of RNA and

proteins through the gene expression process. The gene expression involves a sequence of reactions between various molecules such as DNA, RNA and proteins. A eukaryotic organism contains the complete genome in the nuclei of most of the cells. In this study, we focus on the control factors for the gene expression in the process called transcription.

2.1.1 DNA structure

A DNA molecule consists of two strands, which are holding together by the hydrogen bonding between their bases and form a 3-D structure called double helix [3] as shown in Fig 2.1. DNA sequence has directionality (from 5' end to the 3' end) and in databases such sequences are usually presented in the 5' to 3' directions.

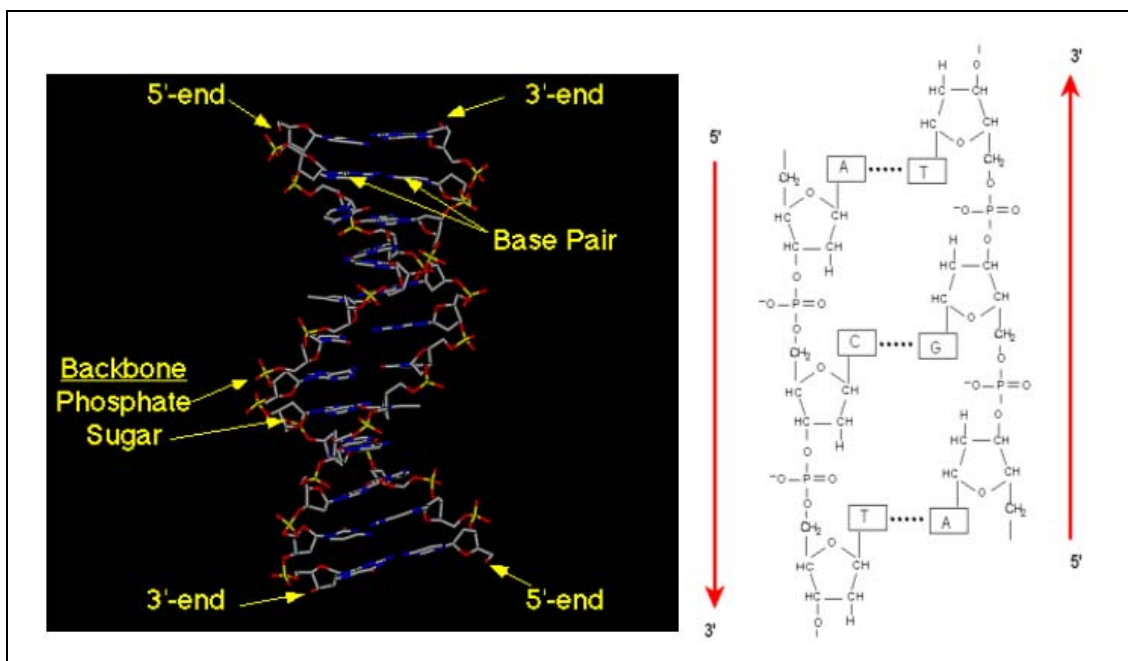


Figure 2.1 Presentation of a double helix structure and chemical compound representation [4]

The basic unit of DNA is a nucleotide, which comprises sugar-phosphate backbone and one of the four bases adenine (A), cytosine (C), guanine (G) and thymine

(T) as illustrated in Fig 2.2. A and G nucleotides (classified as **purines**) contain a pair of fused rings, while C and T (classified as **pyrimidines**) contain only one ring.

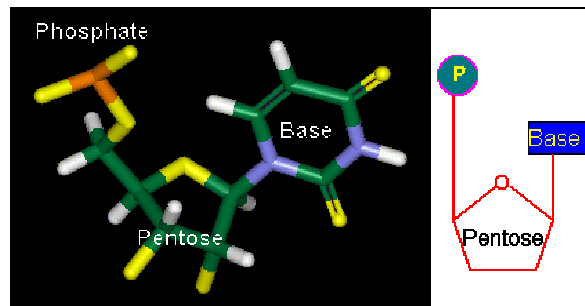


Figure 2.2 Features of nucleotide: phosphate, pentose and base [4]

In the helix strand, double hydrogen bond is formed between T and A in the different strands, while C forms a triple hydrogen bond with G between the strands. Hence, only one strand is used to represent the double strand sequence features, because the opposite strand is **complement** to the other.

Human genome has size of 3×10^9 base pairs (bp) that approximately make 2 meters in length [3]. However, it is presented in a highly compact form of chromosomes through the various levels of packaging [3].

2.1.2 Gene

Genomic sequence consists of different structural patterns, which are also known as genomic features. It includes genes, regulatory elements, repetitive elements, etc., which may have specific functional and biological significance for the functionality of cells.

Genes are regions of DNA sequence that encode information essential for the synthesis proteins and other molecules that are necessary for the correct functioning of

cells. Gene segment may be divided into **regulatory** and **transcribed region**. The regulatory region does not show a clear position relationship relative to the transcribed region. But they are essential for the expression of genes products (peptide or RNA). The transcribed region consists of **exons** and **introns**. Exons encode a peptide or functional RNA. Introns are separators that frequently contain regulatory elements necessary for transcription. Introns will be removed after transcription. The boundary between the exons and introns contain specific signals where splicing occurs. Splicing depends on the condition, which may result in different closely related proteins being expressed. Schematic presentation of a gene segment is shown in Fig. 2.3.

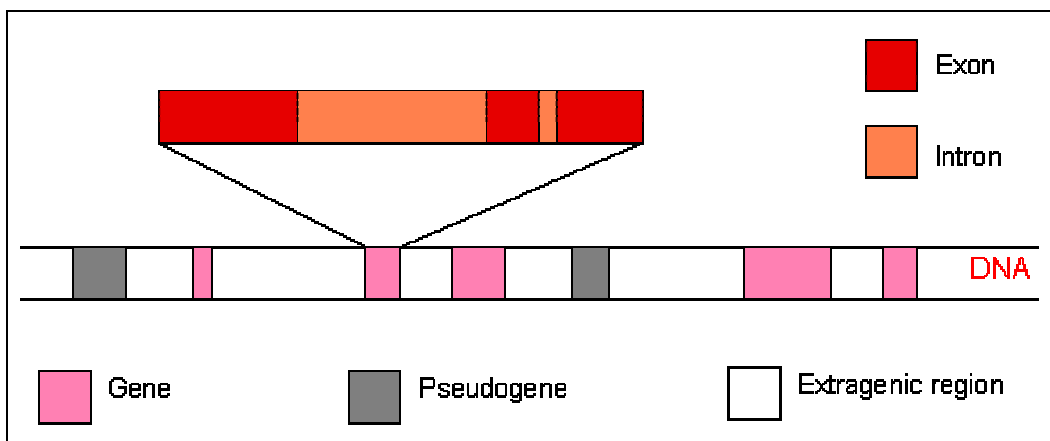


Figure 2.3 General organization of the DNA sequence. Only the exons encode a functional peptide or RNA. The coding region accounts for about 3% of the total DNA in a human cell [6]

2.1.3 Regulatory factors

Transcription of every gene is controlled through different regulatory regions, such as promoters, enhancers and silencer, which perform different functions during gene expression. These regions contain binding sites for various regulatory factors, TFs, which get bound and bind to the available TFBS and in this way regulate gene expression. However, each TF may have alternative binding sites with different affinities depending

on the biological and chemical conditions in the cell. So, gene expression shows different characteristics during different cellular conditions.

TFs are proteins that may have multiple binding sites with different levels of affinity for a TF [2]. The effect that TF may exert to the gene expression is not only determined by the location and orientation of individual TFBS, but also by their context and the relative distances between them and other TFBSs [7].

2.1.4 TF binding sites

TFBSs are small sequence regions consisting of 5-25 bp where TFs bind to regulate and/or initiate transcription. They are present in the promoter region and upstream regulatory sequences. They sometimes show specific pattern with respect to location and orientation within the promoter sequences.

2.1.5 Promoter Fundamentals

A promoter can be considered a DNA segment mainly responsible for gene transcription. The promoter is recognized by RNA polymerase and TFs, which then initiate transcription. Promoters also represent the demarcation region to denote which genes should be used for messenger RNA creation and consequently control which proteins will be produced in a cell.

A promoter could be structurally divided into three parts: core, proximal and distal promoters, according to their positions in the sequences [7].

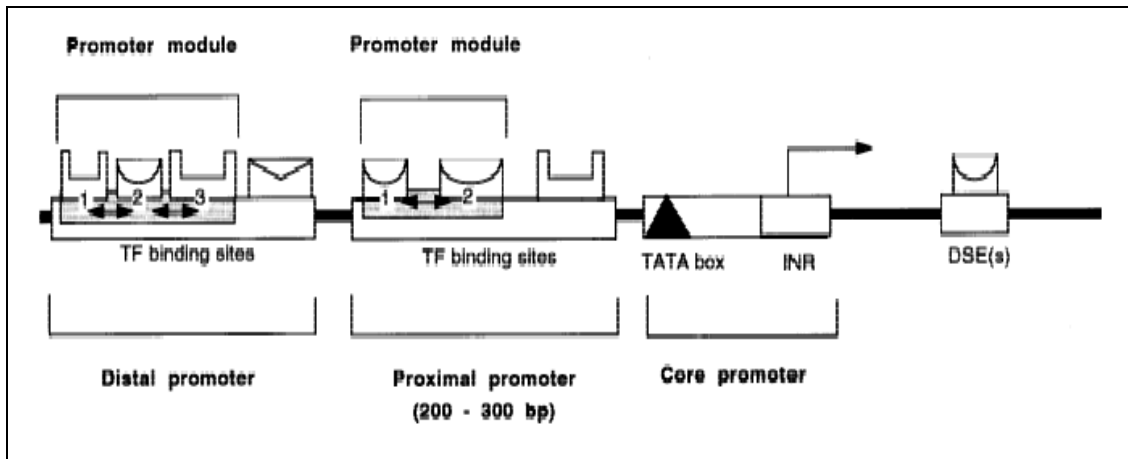


Figure 2.4 A typical structure of promoter showing binding sites and promoter modules. [7]

Core promoter is the promoter segment, which is to determine the precise transcription start site. It is usually located at -35 to +35 region of promoter and contains binding sites of general TFs involved in initiation of transcription like TATA box , Inr (initiator), BRE (TFIIB recognition element) , DPE (downstream core promoter element). Each of these motifs has specific function in the process of transcriptional regulation. However, these elements appear in most of core promoters, but not all.

The proximal promoter is the region, which is in the immediate vicinity of the minimum promoter site (roughly from -250 to +250 nt). The proximal promoter contains the functionally important regulatory controls. CCAAT box is an example of TFBS located in the proximal promoter.

Distal promoter is the region on the DNA upstream of the proximal promoter where regulatory TFs bind. It may be located thousands of bps away from the TSS (Transcription Start Sites). The distal promoter can consist of binding sites for any of TFs.

Enhancer is the DNA regions which are usually rich in TFBSs and/or repeats. They enhance transcription of the responsive promoter independent of orientation and

position. Silencer is also the DNA region far away from the TSS, but it decreases the transcription.

2.1.6 Gene expression and transcription mechanism

Gene expression is the process by which a gene's information is converted into the structures and functions of a cell. It is a multi-step process. Here we only focus on the transcription process. Figure 2.5 shows main gene expression steps.

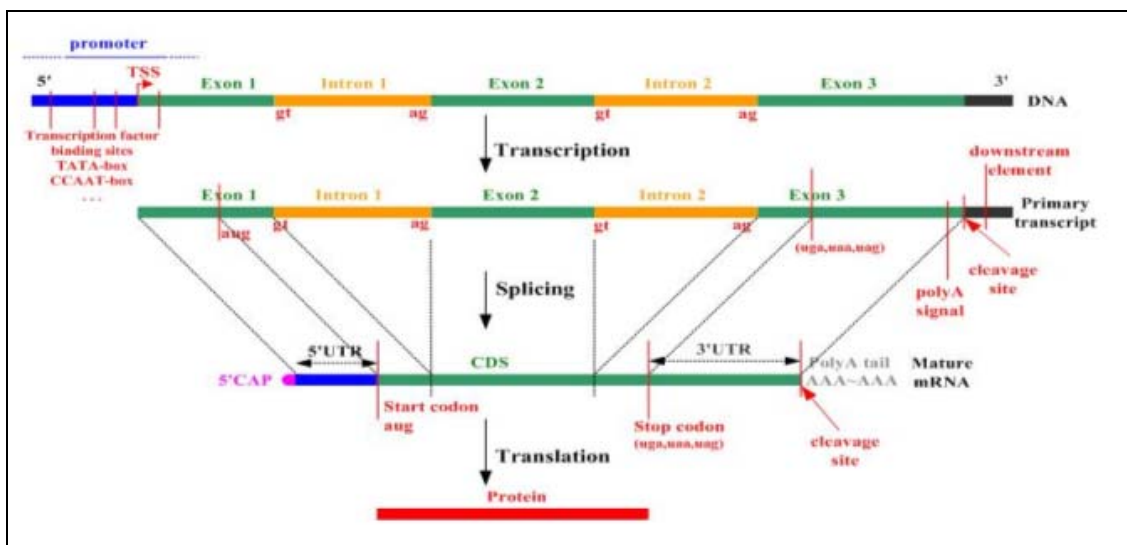


Figure 2.5 Process of Eukaryotic Gene expressions

Transcription: Transcription represents the first stage of gene expression, when a DNA sequence is enzymatically copied by an RNA polymerase to produce a complementary RNA. In the case of protein-encoding DNA, transcription is the beginning of the process that ultimately leads to the translation of the genetic code (via the mRNA as an intermediate product) into a functional peptide or protein.

Basics about Transcriptional control in general:

Understanding the mechanism of gene transcription is essential for us to investigate and explore important parts of the gene control factors. The transcription mechanism involves various proteins (TFs, TAFs (transcription accessory factors), and GTFs (general TFs)), their complexes, and RNA polymerase II, which form an assembly known as transcription initiation complex (TIC) for transcription initiation.

Initiation of transcription requires the enzyme RNA polymerase and TFs. Any protein that is needed for initiation of transcription, but not itself part of RNA polymerase, is defined as TF.

Initially, in transcription initiation requires that the different TFs bind to upstream promoter and enhancer sequences and form a multiprotein complex. Then, this complex directly or indirectly attracts to the core promoter a Polymerase II that is complexed with some GTFs. Transcription is initiated by this initiation complex.

The following is the simplistic model of transcription initiation process.

- TFs get attached to TFBSs in promoters, enhancer or silencer regions [7]. TFs may be activators or repressors to regulate the transcription process.
- TAFs complex with TFIID (transcription factor IID) macromolecule, whose TBP gets bound to the TATA box and thus determines the location of the TSS in the core promoter.
- Polymerase II gets complex with other GTFs and gets bound to the core promoter to form TIC. TIC is the key complex to initiates the transcription.

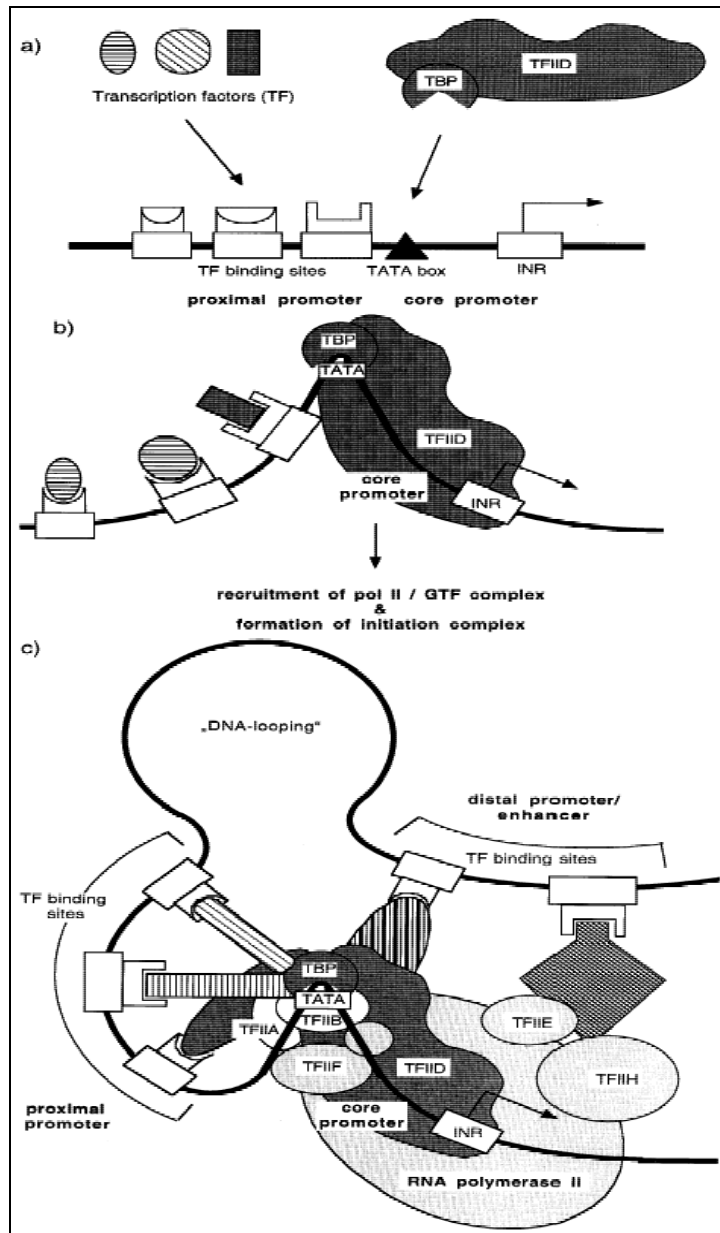


Figure 2.6 Assembly of the activator/promoter complex on the proximal and core promoter region. a) Schematic representation of the proximal promoter with these specific TF binding sites and the core promoter represented by the TATA box (black triangle) and the initiator region (INR). The transcription start site (TSS) is indicated by the angled arrow. b) Binding of the TFs and the TFIID complex (including the TAA box binding protein TBP). TBP binding induces a 90° bend in the promoter DNA. c) Subsequently the polymerase II/GTF complex is loaded to yield the complete initiation complex. [7]

2.2 Bioinformatics

Bioinformatics utilizes the computation technique to explore efficiently the molecular biology in a large-scale fashion. It involves different techniques that enable highly efficient sequence data manipulation and database searches. One of the key challenges of bioinformatics is to handle the voluminous sequence information and to design more efficient analysis tools to manipulate the data [1], so as to enable sequence information into relevant biological knowledge

In our work, motif recognition with heuristic algorithms and promoter structure predication are developed to study the content of promoter in the genomic environment. The content of promoters enables us to generate information to reconstruct transcriptional regulatory networks. Therefore, two key areas that are focus of this study are motif prediction and presentation of the gene correlation information through promoter content.

2.2.1 Motif Prediction

Motif discovery is one of the key problems related to analysis of regulatory regions. In this study we use computational methodology to automatically discover motif families from a set of DNA sequences. We extract the candidate motifs and construct the representation of approximate distribution of such patterns in the set of sequences from which motifs are extracted.

Since 1980s many approaches have been developed for motif discovery attempting to locate regulatory elements. Probabilistic and combinatorial algorithms are dominant methods to determine TF-binding motifs common between the sequences. MEME [9, 10], AlignACE [11] and CONSENSUS [12] are examples of some of the best

known systems for identification of DNA motifs. Many other methods are also known [46]. Here we explain properties of some of these systems.

a) MEME: Expectation Maximization

MEME (<http://meme.sdsc.edu/meme/website/meme.html>) [9, 10] utilizes the finite mixture model (MM) to classify the given data set. MM is a probabilistic model with two parts. One part is the motif model (with probability λ_1) that describes the distribution properties of the motif (with position weight matrix of nucleotide frequencies $\theta_1 = (f_1, f_1, \dots, f_w)$); and the other is the background model (with probability $\lambda_2 = 1 - \lambda_1$) which describes the properties of the background subsequences.

The algorithm in MEME is an extension of the expectation maximization (EM) technique for fitting finite mixture models developed in [9,10]. The EM algorithm makes use of the concept of missing data. Starting from an initial motif, MM iteratively obtains a better motif through the E-step (Expectation step) and the M-step (Maximization step). The E-step calculates the expected log likelihood over the conditional distribution of the missing data. The M-step updates the over model parameters by maximizing them from the probabilistic results of the E-step.

There are some further developments of the algorithm in following MEME. The unsupervised learning of multiple motifs and methods of combining motif match scores have been implemented to enhance the search function and improve the accuracy of the prediction results. Therefore, MEME is considered as superior to the other methods by its prediction accuracy, but has the drawback of taking enormous computation time.

b) AlignACE: Gibbs Sampling

Developed by genomics researchers at Harvard Medical School, AlignAce (<http://atlas.med.harvard.edu/>) employs the Gibbs sampling algorithm that scans non-coding nucleic acid sequences at high resolution for motifs that occur with non-random frequency. This algorithm is built into a multi-level sequence analysis program that highlights gene-specific regulatory elements for further analysis.

Gibbs sampling in statistics is a technique for generating random variables from a marginal distribution indirectly, without having to calculate the density [12]. This approach is based on elementary properties of Markov Chains. Initially, AlignACE obtains the number of occurrences of certain nucleotide M_{kj} in specific motif position by selecting the random locations $\{a_1, \dots, a_n\}$ in different sequences $\{x_1, \dots, x_n\}$. Then it starts iteration by predictive updating and near optimum sampling. The predictive update is to remove certain sequence x_i in the data set and recomputed model M_{kj} . The near optimum sampling is to sample a new random position over the background and obtain the optimum value. AlignAce offers both efficiency and convenience. Its high signal-to-noise ratio preferentially reduces false positives in the program output, while iterative masking uncovers multiple, distinct sequence motifs within a single data set.

c) CONSENSUS: Matrix of consensus pattern

CONSENSUS is a matrix-based pattern discovery for DNA or protein sequence sets [12]. It uses greedy multiple alignment algorithm to search for a motif alignment, which maximizes the information content score of the model. The CONSENSUS first randomly selects one sequence as start sequence, and extracts subsequences with fixed

length l as single pattern motifs; then it catches the right signals of the motif model through the top Q (where Q is a user-designated parameter, the default Q in CONSENSUS is 1000) pair-wise pattern similarities between this start sequence and one of the remaining sequences; after that, it iteratively assembles the top Q signals into multiple similarities by adding more and more pattern instances from different sequences with a greedy selection algorithm. The time complexity of this algorithm is $O(nm^2 + Qn^2ml)$, where n is the number of sequences in the data set, and m is the average length of sequences.

2.2.2 Graphical presentations of various biological information

Graphical presentation is an effective and efficient approach to describe the biological information or other complex information where a lot of interconnections appear, or when information is complex. For presenting complex information graphically, we can use, for example, color, size, shape, line thickness, line forms and types, arrow, and position to present various attributes of information. In our work, one of the objectives is to present the association of TFs and motifs and their target genes in a graphical format.

Different ways of graphical presentations have been developed and integrated into various bioinformatics systems to enhance interpretation of the results. For example, JASPAR [13], DTFAM [14], CellDesigner [41], and UCSC browser [42] do provide a vivid graphical description of relevant biological information. In what follows we present several systems that use various forms and approaches to present graphic information suited to the problems of their particular interest.

a) JASPAR

JASPAR [13] (<http://jaspar.cgb.ki.se/>) is an open-access database of annotated, high-quality, matrix-based TFBS profiles for multi-cellular eukaryotes. It presents the basic information as the so-called sequence logo [52]. An example is depicted in Fig. 2.7.

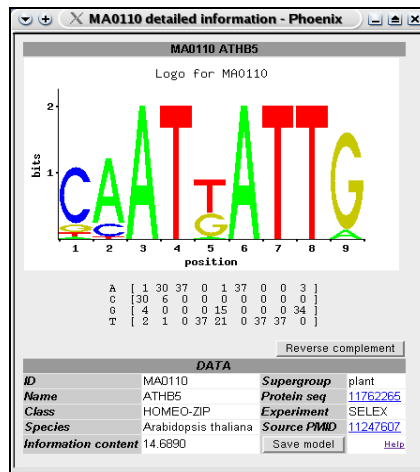


Figure 2.7 Matrix based TF profile [13]

It utilizes the SockEye [63] visualization tool to present the TF profile, as shown in the following diagram. In the JASPAR, logos are a visual representation of a profile, based on Shannon information content [13], in which maximal conservation amounts to a information content of 2 bits for a single position. There is other information provided in the JASPAR interface, such as ID, class, supergroup, etc.

b) DTFAM (DRAGON TF ASSOCIATION MINER)

DTFAM [14] (http://research.i2r.a-star.edu.sg/DRAGON/TFAM_v2/index.html) is a web-based system to provide information about potential association of TFs with terms from the four well-controlled vocabularies so as to help biologists infer unusual

functional associations. It was developed in Institute for Infocomm Research, Singapore. It uses the Graphviz software to generate graphical presentation.

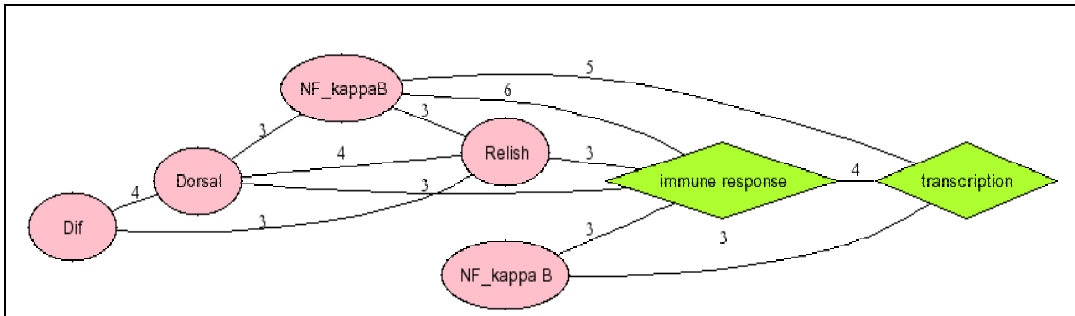


Figure 2.8 Association of e different terms defined in PubMed documents. Documents were collected based on query “antimicrobial toll”. Antimicrobial peptides are important component of innate immune system in vertebrates. Gene with produce them are mainly controlled through the toll-like receptor pathway of which NF-kappaB is one of the key regulators. Text-mined information conveniently presents such associations.

DTFAM analyses the connections (associations) between the all terms and expressions found in the selected documents and generates one or more association map networks. The association of vocabularies is based on their co-occurrence in the same PubMed document. The nodes of the generated graphs represent the terms from the selected vocabularies. TF names are presented by the ellipsoidal nodes with yellow background. Diseases are represented by ellipsoidal nodes with gray background. Terms from gene ontology (GO) categories are represented by rhomboidal shapes with biological processes having green background, molecular functions with nodes having light blue background, while cellular components are represented with nodes having magenta background. All nodes provide links to a set of related PubMed documents with color-marked terms to allow for user’s inspection and assessment of the relevance of proposed associations. This system has made this task easier for the user by providing links to the documents used, and we also color-highlighted the terms used in the analysis.

c) CellDesigner

CellDesigner is a structured diagram editor for illustrating gene-regulatory and biochemical networks. Networks are drawn based on the process diagram, with graphical notation system proposed by Kitano [41], using the Systems Biology Markup Language (SBML), a standard for representing models of biochemical and gene-regulatory networks. Networks are able to link with simulation and other analysis packages through Systems Biology Workbench (SBW).

A process diagram is a state transition diagram with complex node structures. It consists of two classes of vertexes and edges, which represents the state of the entities. In this software, the process of the diagram graphically represents the state transitions of the molecules involved, which could illustrate the interactions and associations of the bindings for the molecular species more intuitively.

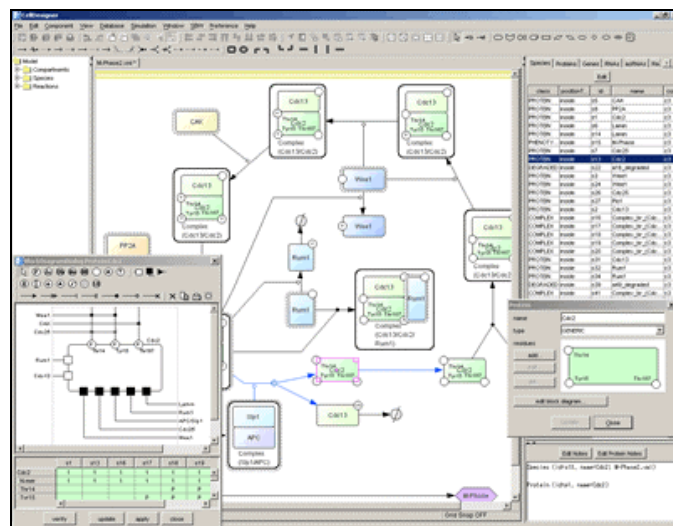


Figure 2.9 The snapshot of the CellDesigner 3.0 [41]

d) ENSEMBL

Ensembl (<http://www.ensembl.org>) is a software system that produces and maintains automatic annotation on selected eukaryotic genomes. It is a joint project, which is developed by European Bioinformatics Institute (EBI) and the Wellcome Trust Sanger Institute (WTSI).



Figure 2.10 Snapshot of the multicontigview expression in ENSEMBL [42]

The most prominent annotation to the website is multicontigview, which allows regions of genome sequence from multiple species to be viewed aligned to each other. Besides making these alignments accessible to a wider audience, multicontigview allows the alignment of as many genomes as desired and is able to show in a single display both DNA similarity and putative ortholog relationships. Multicontigview is complementary to the display of regions of conservation in contigview. Whereas the latter is useful to

identify important regions in a single genome, multicontigview allows researchers to compare annotation between genomes to look for places where annotation may be missing.

Comment on the previous work on graphical presentation

There is no universal solution to simply present complex biological information. Different applications have been developed for the specific bioinformatics problems being more suited to the problem of interest. Systems equipped by graphical presentation of some aspects of information usually enhance human-readability and comprehension, with clear and unambiguous graphical presentation. However, most of the software systems lack interaction and flexibility that can enhance usability and can help easier interpretation of complex biological knowledge.

2.2.3 Graph drawing packages and applications

Graph drawing is the approach to provide the graphical presentation. In mathematics and computer science, graphs could be understood as the representation in form of dots (nodes, vertices) and edges (arcs, links) connecting of the dots. Graphs can be classified as directed and undirected, depending on whether an edge is assigned an orientation [47]. Presentation of information via graphs is studied in computer science and includes graph theory, geometry, topology, visual languages, visual perception, information visualization, computer-human interaction, and graphic design [47]. It utilizes topology and geometry to derive visual and haptic representations from a dataset.

Graph drawing is suitable for those applications where it is crucial to visualize structural information in visual graphic format. Indeed, advances in graph drawing are the

key factors in such technological areas as Web applications, E-commerce, VLSI circuit design, information systems, software engineering, computational cartography, bioinformatics, and networking. Therefore, great effort has been spent on algorithms and applications on the geometric representation of graphs and networks. Thus, significant progress has been made in development of software to visualize the graph and networks.

The softwares are listed briefly as under:

- Graphical library, such as OpenGL [60] and GD [48]
- Programming languages, such as SBML (Systems Biology Markup Language) [59] and VRML (Virtual Reality Modelling language) [49],
- The computer aided design tools, such as SolidDesigner [61], AutoCAD [62], etc.

There are no universal solutions for various types of applications and all different aspects that users may want to have, so these libraries, languages and modeling tools are designed to cater for the different purposes.

Currently, very few graphical drawing applications cater for expression of biological information. Our study focuses on the graphical drawing applications in bioinformatics, especially for representation of information related to transcription regulation. However, our objective is not to develop the libraries or software to represent the biological data, but to utilize the existing graphical drawing languages and libraries to generate suitable graphical presentation to visualize such complex biological information. The following paragraphs explain some widely used library packages and languages used in bioinformatics.

1. OpenGL

OpenGL (Open Graphics Library) is a software interface to graphics hardware, which is governed by the OpenGL Architecture Review Board (ARB) [60]. OpenGL is the premier environment for developing portable, interactive 2D and 3D graphics applications, which involves a set of procedures and functions to interface with the hardware.

Since released in 1992, OpenGL has become the industry's most widely used and supported 2D and 3D graphics application programming interface (API), bringing thousands of applications to a wide variety of computer platforms. OpenGL fosters innovation and speeds application development by incorporating a broad set of rendering, texture mapping, special effects, and other powerful visualization functions. The well-specified OpenGL standard has language bindings for C, C++, Fortran, Ada, and Java. It can be supported on UNIX workstations, Windows 95/98/2000/NT and MacOS PC. It is a useful and important tool for developers to access geometric and image primitives, display lists, model transformations, etc.

2. GD Library

GD is an open source code library to create images [48]. GD is developed in C language and it is also has interface with Perl, PHP and other languages. GD can create PNG, JPEG and GIF images, among other formats. It is used to generate charts, graphics, thumbnails, etc. The GD is common and popular package for the web-based graphical application because PNG and JPEG formats generated by this library are commonly accepted formats for inline images by most browsers. Thus, this library package is an

essential component to be incorporated in generation of visual tools for representing biological information in the web-based applications, like in databases.

3. VRML (Virtual Reality Modeling language)

VRML is a standard file format for representing 3-dimensional (3-D) interactive vector graphics, designed particularly with the web application. This language is conceptualized in 1994, and developed by a lot of researchers [49,50]. It allows to build a series of visual images into web settings with which a user can interact by viewing, moving, rotating, and otherwise interacting with an apparently 3-D scene.

VRML is a tool that enables representation of a 3-D polygon with effects like surface color, image-mapped textures, transparency and so on. VRML when installed, facilitates URLs (web browsers) to convert a text file containing information in terms of vertices and edges (co-ordinate information) of a 3-D polygon to graphical image. Moreover, VRML allows user to dynamically change or add animations, sounds, lighting, and other aspects of the virtual world. Therefore, it has applications in creation of graphical tools in the domain of bioinformatics. For example, SockEye [63] and ENSEMBL [44] utilize this tool.

4. Graphviz

Graphviz [37] is open source graphic visualization software developed by AT&T. Different from the previously mentioned softwares and libraries, Graphviz focuses on the applications of the graph layout, which is to visualize the structural information as diagram of abstract graphs and network.

Graphviz consists of implementations of various common types of graph layout. These layouts can be used via a C library interface, stream-based command line tools, graphical user interfaces and web browsers. It possesses the characteristics, which allows graph manipulation and supports for a wide assortment of graphical features and output formats. With this functionality, programmers can query, modify and display graphs using high level language like Java, Perl etc.

Many bioinformatics applications employ Graphviz to produce graph layouts, in order to assist biologists to understand complex domain information or to perform the interpretation of the data. Protein Interaction Extraction System (PIES) [51] and DTFAM [14] are the examples of complex applications that utilize this software in bioinformatics domain.

The objects analyzed in bioinformatics are complex biological entities, structures and processes. No universal solution in graphics representation can express the complexity of gene and protein sequence information effectively. Thus, we have attempted different approaches to cater for the different needs that biologists have in relation to a particular topic like transcription regulation. In our work, we utilized the GD library to visualize the TFBSs near the TSS in the database, and we have made use of the Graphviz package to present the networks between the PEs and the genes.

Chapter 3 Ab-initio Motif Discovery

3.1 A broader context of motif discovery: Gene Finding

Motif discovery is one of the important steps in understanding the genome of a species once it has been sequenced. It can be used in gene finding, which is the area of bioinformatics that is concerned with algorithmically identifying stretches of DNA sequence that are biologically functional and represent domains that are transcribed. This especially includes protein-coding genes, but may also include other functional elements such as RNA genes and regulatory regions.

Determination whether a sequence is functional should be distinguished from determining the function of the gene or its product. The latter still demands in vivo experimentation through gene knockout and other assays, although current genomics and bioinformatics are making it increasingly possible to predict the function of a gene based on its sequence alone. Today, with comprehensive genome sequence and powerful computational resources, motif discovery has been redefined as a largely computational problem. The comprehensive computation algorithms are useful tools to prepare the data for the graphical presentation in our study.

There are a number of computational techniques which have been proposed to solve the gene finding problem. Generally they could be classified into three different groups:

a) Extrinsic Approach

The target genome is searched for sequences that are similar to extrinsic evidence in the form of the known sequence of a messenger RNA (mRNA) or protein product.

Given an mRNA sequence, it is possible to derive a unique genomic DNA sequence from which it was transcribed. When a protein sequence is available, a family of possible coding DNA sequences can be derived by reverse translation of the genetic code. Once candidate DNA sequences have been determined, it is a relatively straightforward algorithmic problem to efficiently search a target genome for matches, complete or partial, exact or inexact. BLAST is a widely used system designed for this purpose. [15]

b) ab initio Approach

Ab initio gene finding is a systematically searched methodology for certain signs of protein-coding genes in genomic DNA sequences. These signs can be broadly categorized as either signals, specific sequences that indicate the presence of a gene nearby, or content, statistical properties of protein-coding sequence itself. Ab initio gene finding might be more accurately characterized as gene prediction, since extrinsic evidence is generally required to conclusively establish whether a putative gene is functional.

c) Comparative Genomics Approach

This approach is based on the principle that the forces of natural selection cause genes and other functional elements to undergo mutation at a slower rate than the rest of the genome, since mutations in functional elements are more likely to negatively impact the organism than mutations elsewhere. Genes can thus be identified by comparing the genomes of related species to detect this evolutionary pressure for conservation.

Our research focuses on algorithm development and specific computational methods for the ab-initio motif detection in DNA and protein sequences. However, ab initio gene finding in eukaryotes, especially complex organisms like humans and mouse, is considerably more challenging for several reasons:

First, the promoter and other regulatory signals in these genomes are more complex and less well-understood than in prokaryotes, making them more difficult to reliably recognize.

Second, splicing mechanisms employed by eukaryotic cells mean that a particular protein-coding sequence in the genome is divided into several parts (exons), separated by non-coding sequences (introns). (Splice sites are themselves another signal that eukaryotic gene finders are often designed to identify.) For example, a typical protein-coding gene in human might be divided into a dozen exons, each less than two hundred base pairs in length, and some as short as twenty to thirty. These splicing mechanisms affect the accuracy of gene prediction significantly.

Therefore, ab-initio gene finders for both prokaryotic and eukaryotic genomes typically use complex probabilistic and computational linguistic models, especially heuristic algorithms, in order to combine information from a variety of different signal and content measurements. So, in our work, the heuristic algorithm and the local alignment method are implemented to construct the motif discovery system.

3.2 Heuristic Algorithms in Motif Discovery

Two fundamental goals in computer science are searching algorithms with hopefully good run times and with hopefully good or optimal solution quality. A heuristic

is an algorithm that optimizes both of these goals; for example, it usually finds pretty good solutions within a reasonable run time. It could be one of the best computational methodologies to analyze the large scale sequence data accurately within an optimum time.

Generally, biological sequences which belong to a group of functionally related genes or proteins, usually contain a number of sequence patterns which are shared among many and sometimes all members of the functional group. A typical example represents promoters of a group of co-expressed or co-regulated genes which contain many common transcriptional regulatory elements which also share similar positional organization (order and distances of transcriptional elements). For this project we propose to use a set of heuristic algorithms to determine the most consistent set of regulatory patterns in functionally related groups of biological sequences (either DNA or proteins).

In my work, the heuristic methods, genetic algorithm (GA) and expectation maximization (EM), are implemented to achieve both the speed of extraction and consistency of extracted motif groups. These methods can find direct application in discovery of TFBSs, and more generally, in determination of functional patterns in DNA/RNA and in proteins.

3.2.1 Expectation Maximization (EM) Algorithm

EM algorithm is an algorithm to estimate maximum likelihood of parameters in probabilistic models, where the model depends on unobserved (latent) variables. EM alternates between performing an expectation (E-step), which computes the expected value of the latent variables, and a maximization (M-step), which computes the maximum

likelihood estimates of the parameters given the data and setting the latent variables to their expectation [16].

It can be shown that EM iteration does not decrease the observed data likelihood function, and that the only stationary points of the iteration are the stationary points of the observed data likelihood function. In practice, this means that an EM algorithm will converge to a local maximum of the observed data likelihood function.

In our work, EM is used to estimate the probability density of the most popular patterns within a set of DNA sequences. The optimal motifs are predicted with pattern matching score function and the population of the motifs among the sequences. EM algorithm iteratively augments the motif data by guessing the values of the optimal score and population with the sequence, and then **re-estimates the parameters** by assuming the “best” value for the motif group. In order to model the probability density of the data effectively, **most likelihood function** was implemented to choose the initial value that has highest converged likelihood value [17, 18]. The threshold coefficient for **information content (IC)** has been applied to improve the efficiency and accuracy of the search approach.

a. E-step (Expectation): Computing the $Q(\theta_{n+1}|\theta_n)$

E-step computes the expected likelihood for the complete data (Q) where the expectation is taken from the computed condition distribution θ_n of the latent variables θ_{n+1} (i.e., the hidden variables) given the current settings of parameters and observed (incomplete data).

Q is the expected likelihood for the complete data set; in our systems, it is defined as the **optimal coefficient** between the **IC** and the **size of motifs group**. The IC represents the consensus of the group of motif patterns, because we believe the motifs with same biological functionality possess the similar pattern. Besides the similarity of the pattern, the large population is also one of the important factors that we are interested in. We believe, the more frequently the similar patterns appear the sequences, the stronger the biological signal they represent.

To obtain the Q factor among the sequences data, position weight matrix (PWM) is formed according to the group of consensus patterns observed, and the Q can be derived from the PWM. PWM is the pattern matrix, which enables representing nucleotide low/high affinity in different positions. The following example illustrates PWM of one group (6) motif patterns.

Table 3.1: Align pattern extracted from sequences

Motif	1	2	3	4	5	6	7	8	9	10
#1	A	G	A	T	G	G	A	T	G	G
#2	T	G	A	T	T	G	A	T	G	T
#3	T	G	A	T	G	G	A	T	G	G
#4	A	G	A	T	T	G	A	T	C	G
#5	T	G	A	T	G	G	A	T	T	G
#6	T	G	A	T	G	G	A	T	T	G

Conversion of PWM with the aligned patterns as Table 3.2

Table 3.2: PWM of the align motifs

Nucleotides	1	2	3	4	5	6	7	8	9	10
A	2	0	6	0	0	0	6	0	0	0
C	0	0	0	0	0	0	0	0	1	0
G	0	6	0	0	4	6	0	0	3	5
T	4	0	0	6	2	0	0	6	2	1

Normalized PWM obtained from the aligned patterns

Table 3.3: Normalized PWM

Nucleotides	1	2	3	4	5	6	7	8	9	10
A	0.33	0	1	0	0	0	1	0	0	0
C	0	0	0	0	0	0	0	0	0.17	0
G	0	1	0	0	0.67	1	0	0	0.50	0.83
T	0.67	0	0	1	0.33	0	0	1	0.33	0.17

The IC, which is the similarity of the patterns, could be translated into the PWM mathematically.

$$IC = \sum_{j=0}^4 p_{i,j} \times \left(\sum_{i=1}^L \sum_{j=1}^4 (p_{i,j} \times \log(p_{i,j})) \right) + \sum_{i=1}^L \sum_{j=1}^4 (p_{i,j} \times \log(p_{i,j})) \quad (1)$$

$$p_{i,j} = \frac{P_{i,j}}{\sum_{j=0}^4 P_{i,j}} \quad (2)$$

$$Q = \frac{1}{G} \times IC \quad (3)$$

G: the total number of sequences

$P_{i,j}$: the element of the PWM

$p_{i,j}$: the element of the normalized PWM

i, j : column and row for the corresponding PWM

The element of normalized PWM could be obtained from the raw one with the formula (3). Then the formula (1) will determine IC of the motif group. The Q factor is the optimum value with the size of the motif group (G) and their similarity (IC). It represents the expected likelihood of the consensus motifs among the sequences.

b. M-step (Maximization): Maximizing $Q(\theta_{n+1}|\theta_n)$ with respect to θ_n

With the expected Q factor obtained from the E-step, the M-step re-estimates all the parameters by maximizing it. The corresponding new estimate ($\theta_{n+1}|\theta_n$) is expected to lie closer to the location of the nearest local maximum of the likelihood. For our analysis, the new group of patterns is obtained regarding the PWM set as to improve the similarity of the motif patterns with the defined $\theta_{threshold}$.

$$\theta^* = \max Q(\theta | \theta_n) \quad (4)$$

$$\theta^* > \theta_{threshold} \quad (5)$$

$$\theta^* = \frac{1}{L} \sum_{i=1}^L (p_i(j)) \quad (6)$$

L: length of the motif

The most similar patterns are extracted to construct the new group of motif, with the score θ^* of the patterns regarding to the normalized PWM (Table 3). Then the score of the pattern would be obtained by comparing the pattern with the PWM.

With the formula (4), to maximize the IC, the pattern with the best score will be chosen to construct the next group motif to proceed to the next E-step. The following examples illustrate how the patterns are converted to θ^* according to the normalized PWM mathematically.

$$(AGATGGATGG) \theta^* = (0.33 + 1 + 1 + 1 + 0.67 + 1 + 1 + 1 + 0.5 + 0.83)/10 = 0.833$$

$$(ACTGGGATCT) \theta^* = (0.33 + 0 + 0 + 0 + 0.67 + 1 + 1 + 1 + 0.17 + 0.17)/10 = 0.434$$

$$(TCGATCTACT) \theta^* = (0.67 + 0 + 0 + 0 + 0.33 + 0 + 1 + 0 + 0.17 + 0.17)/10 = 0.234$$

In order to maximize the $Q(\theta|\theta_n)$, the threshold of the score (S_{th}) has been implemented to preventing the patterns with low score being extracted into the next

group of motifs. For example, if the S_{th} was set to 0.85, the score of three patterns discussed previously are below the threshold, therefore, they would not be chosen. The threshold value is very important to maintain the expected likelihood Q value during the search, and affect the accuracy of prediction.

Another parameter ‘zero-elimination’ is applied on the M-step to enhance the maximization of the Q factor. This parameter will improve searching for the Q effectively and enhance the searching speed. The zero-elimination is used to eliminate those patterns containing the nucleotides, whose $P_{i,j}$ is equal to zero in the PWM. For example for PWM from Table 3.3, pattern (ACATGGAGG) can not be chosen, because second nucleotide C in the motif is zero in the normalized PWM.

c. Initialization Function and iteration

The EM algorithm has a general convergence property via the Jensen’s inequality [19]. Simply speaking, it can be shown that the Q is improved each iteration of M-step. But EM algorithm is a hill-climbing approaching, thus it can only be guaranteed to reach a local maxima.

However, in the biological data, multiple maxima, pseudo-motifs, exist among the sequences. It is often required to identify the global maxima within the multiple local ones to obtain the actual motifs. In order to reach the global maxima, it depends on where the start point is, therefore, the concept of K operator is induced to carefully optimize the initial condition.

$$K = \max \left(\sum_{i=1}^L \sum_{j=1}^4 (p_{i,j} \times \log(p_{i,j})) \right) \quad (7)$$

The algorithm randomly initiates different PWM, and chooses the highest converged one as the initial value according to the K operator. This initial value selection with the heuristically likelihood function can locate a rough region where the global optima exists, and then starting with this Q value, expectation and maximization method are implementing to search for a more accurate optima.

The iteration is controlled by the complete likelihood coefficient ζ , which is assumed to be known. Overall expectation and maximization steps would be stop once the likelihood reaches the level of ζ . However, the assumed ζ value might not be practical for all the cases. Because the EM is heuristic algorithm, if starting point K is too low to achieve ζ , the search of patterns will become extremely slow or fall into one infinite loop. To prevent this condition happening, in our system, one iteration parameter is applied to stop the iteration once the number of iterations exceeds certain threshold. So, the program will re-initiate another approximate region for search until it can locate the motif group, which could meet the criteria of the complete likelihood of ζ .

Pseudo-code EM

Choose initial PWM randomly

Repeat EM

 Estimate the likelihood Q factor from the PWM

 Maximize $Q(\theta_{n+1}|\theta_n)$ respecting to θ_n with threshold S_{th}

 Constructe the new PWM with $Q(\theta_{n+1}|\theta_n)$

Until terminating condition (see below)

Terminating condition

- Budgeting: allocated computation time used up
- A motif group is found that satisfies minimum criteria
- Combinations of the above

3.2.2 Genetic Algorithm (GA)

GA is a search technique used in computer science to find approximate solutions to combinatorial optimization problems. GA is a particular class of evolutionary algorithms that use techniques inspired by evolutionary biology such as inheritance, mutation, selection, and recombination (or crossover).

GA is typically implemented as a computer simulation in which a population of abstract representations of candidate solutions to an optimization problem evolves toward better solutions. So GA is a population heuristics [20]. The heuristic evolution starts from a population of completely random individual motifs and happens in generations. In each generation, the fitness of the whole motif population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), modified (mutated or recombined) to form a new generation.

In our research, only gene patterns among the sequences data, which are fittest, will reproduce and create a new population, and eliminate the other vice versa. This is performed in the second step (Selection). The idea behind is that "good" sections of the parents will combine to produce even fitter children in the Crossover step. Although many of the children created in this way will not be sufficiently successful to survive the next selection, some will. Last, the survivors will continue mutating to enlarge the fitness function to pass the next selection.

a. Fitness Function

A **fitness function** Q is a particular type of objective function that quantifies the optimality of a solution in a GA so that that particular solution may be ranked against all the other ones. In our work, the fitness function is represented by the the **optimal coefficient** between the **IC** and the **size of motifs group**, which is identical to the Q factor in the EM model.

Another parameter, **nucleotide mismatch**, is induced to measure the fitness of the motif group. Nucleotide mismatch indicates the number of nucleotides different from the current reference motif sequence which the algorithm will tolerate while grouping motif sequences. Its function is similar as the threshold coefficient S_{th} in EM, which eliminates the motif patterns with low score (high mismatch).

b. Selection

Selection is biased towards elements of the initial generation which have better fitness, though it is usually not so biased that poorer elements have no chance to participate, in order to prevent the solution set from converging too early to a sub-optimal or local solution.

For individual, the less mismatch pattern possesses comparing to the reference motif, the fitness score is higher. Considering the population of pattern (gene) with associated fitness, the mean-fitness is obtained from the population.

$$\bar{Q} = \frac{1}{N} \sum_{i=1}^N Q_i \quad (8)$$

Every individual pattern will be copied to the new population, at frequency proportional to its fitness (relative to the average fitness). For example, if the average

fitness is 5.76, and the fitness of an individual pattern is 20.21, and then we have $20.21/5.76 = 3.51$. This individual pattern will be duplicated 3 times and also it will have another probability of 0.51 to have one more copy in the new population. On the other hand, the pattern with low fitness score has low probability to duplicate itself in the selection section. With these concepts, the size of the population changes dynamically to converging to the high fitness pattern in our implementation.

c. Crossover

Crossover (or recombination) operation is performed upon the selected population. In our GA has a single tweakable probability (0.85) of crossover, which encodes the probability that two selected patterns will actually react. A random number between 0 and 1 is generated, and if it falls below the crossover probability, two points are swapped on the parent patterns; otherwise, the two parent patterns are propagated into the next generation unchanged. Crossover results in two new child patterns, which are added to the second generation pool. This process is repeated with different parent patterns until there are an appropriate number of candidate solutions in the second generation pool.

In our implementation we use a two-point crossover, where we randomly select two positions in parent patterns, swap the nucleotides in the selected position between the two parents pattern. The following diagram illustrates the process of crossover when the probability triggers the change.

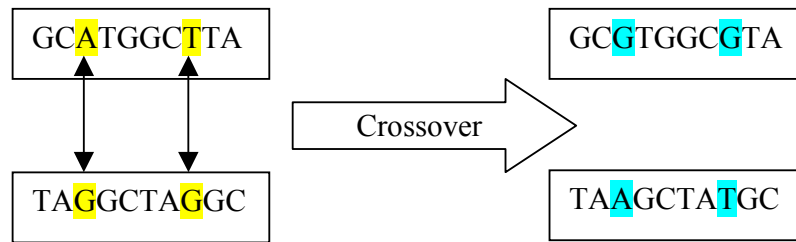


Figure 3.1 Features of two point crossover

d. Mutation

Mutation is to create new offspring pattern, which is controlled by a fixed, very small probability (0.008) of mutation (P_m). A random number between 0 and 1 is generated; if it falls within the P_m range, the new pattern is obtained by randomly altering bits in the parent pattern. It is an element to generate new offspring to maintain the divergence in the population search process. The following example demonstrates how the mutation function is applied.

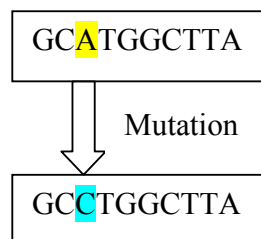


Figure 3.2 Features of one point mutation

Functionality of crossover and mutation in heuristics

The crossover and mutation operators allow the GA to avoid local minima by preventing the population of motifs from becoming too similar to each other, thus slowing or even stopping evolution. This is the reason that we choose a random (or semi-

random) population as starting, instead of one fittest of the population, in generating the next ones.

Pseudo-code GA

Choose initial pattern population

Repeat

 Evaluate the individual fitness of a certain proportion of the population

 Select best-ranking individuals to reproduce

 Mate pairs at random

 Apply crossover operator

 Apply mutation operator

Until terminating condition (see below)

Terminating conditions often include:

- Fixed number of generations reached
- Budgeting: allocated computation time used up
- An individual is found that satisfies minimum criteria
- The highest ranking individual's fitness is reaching or has reached a plateau such that successive iterations are not producing better results anymore.
- Combinations of the above

3.2.3 Statistical Approaches

Building an accurate predictive motif model is essential to be able to differentiate likely motifs from the target group from spurious ones. This is an important step towards understanding gene regulation in the computation biology, as motifs could be real TFBSs. Therefore, statistical approaches are induced to enhance the capabilities to filter out spurious patterns. In our research, Hidden Markov Model [21] and statistical measures, such as P-value and E-value [22, 23], are implemented in the system as the measures of the statistical significance of motifs.

a. Hidden Markov Model (HMM)

In our work, hidden Markov model (HMM) is used to statistically describe a background sequence. This statistical description can be used for sensitive and selective motif search.

HMM is a probabilistic model composed of a number of interconnected states, each of which has an observable output [24], for example, the motif in our case. Transitions among the states are governed by a set of probabilities called transition probabilities. In a particular state an outcome or observation can be generated, according to the associated probability distribution. It is only the outcome that is visible to an external observer, but not the state, and therefore states are “hidden” to the outside; hence the name Hidden Markov Model.

In order to define an HMM completely, following elements are needed.

- The number of states of the model, N .

- The number of observation symbols in the alphabet, M .
- A set of state transition probabilities $\Lambda = \{a_{ij}\}$.

$$a_{ij} = p\{q_{t+1} = j \mid q_t = i\}, 1 \leq i, j \leq N, \quad (9)$$

where q_t denotes the current state.

Transition probabilities should satisfy the normal stochastic constraints,

$$a_{ij} > 0 \text{ and } \sum_{j=1}^N a_{ij} = 1, \text{ where } 1 \leq i, j \leq N \quad (10)$$

- A probability distribution in each of the states, $B = \{b_j(k)\}$.

$$b_j(k) = p\{o_t = v_k \mid q_t = j\}, 1 \leq j \leq N \text{ and } 1 \leq k \leq M \quad (11)$$

where v_k denotes the k^{th} observation symbol in the alphabet, and o_t is the current parameter vector.

Following stochastic constraints must be satisfied.

$$b_j \geq 0 \text{ and } \sum_{j=1}^N b_j = 1, \text{ where } 1 \leq j \leq N \text{ and } 1 \leq k \leq M \quad (12)$$

- The initial state distribution, $\pi = \{\pi_i\}$, where,

$$\pi_i = p\{q_t = i\}, 1 \leq i \leq N \quad (13)$$

Therefore, the compact notation is used, $\lambda = \{\Lambda, B, \pi\}$, to denote HMM with discrete probability distributions.

The discrete HMM is implemented to generate one background sequence to determine the probability that the predicated motif appears in the model sequence. Hence, in our work, the states are defined as:

- Initial nucleotide state distribution, π , is generated randomly by the system.
- The order of observation in HMM, k , is defined by the user. In our work, it is represented the length of motif, which can predict the next nucleotide type possibility.
- State transition probabilities a_{ij} , is obtained from the nucleotide possibility distribution of the foreground target sequences or specific sequence, which is input target background sequence defined by user.
- Distribution state, $b_j(k)$, is the nucleotide appearance possibility with the known k order motif.

With the clear properties definition, the HMM table could be generated. Hence, Table 3.4 is the illustration of the 2nd order HMM table, from which the transition state of next nucleotide could be predicted.

Table 3.4: Normalized PWM

	A	C	G	T
AA	0.186074	0.298308	0.322131	0.193487
AC	0.18435	0.297301	0.322757	0.195593
AG	0.186168	0.300044	0.320398	0.19339
AT	0.185172	0.299291	0.321089	0.194448
CA	0.18658	0.29839	0.321971	0.193059
CC	0.184659	0.298784	0.322031	0.194526
CG	0.186991	0.298732	0.320724	0.193554
CT	0.185396	0.299841	0.321248	0.193514
GA	0.185422	0.297259	0.322015	0.195304
GC	0.186125	0.297871	0.32137	0.194633
GG	0.185399	0.299894	0.320885	0.193822
GT	0.186053	0.298779	0.321261	0.193908
TA	0.185587	0.298737	0.31935	0.196326
TC	0.185115	0.299285	0.32076	0.19484
TG	0.185106	0.299598	0.321791	0.193504
TT	0.186112	0.299274	0.322004	0.192609

The background sequence enhances the motif searching features, in term of sensitivity and specificity. One of the parameters, extinction ratio of the motif in the target and background sequences, is the control factor to distinguish the validity of the motif in the background sequences. Therefore, it can statistically eliminate those spurious patterns, which are extracted in the heuristic algorithm. Moreover, the state transition probabilities, which are generated from user target background sequence, could be specific for certain groups of motifs.

b. Statistical and Analytical Measures

With the background sequence generated from the HMM, it is interesting and important to express the motif's significance. Therefore, some statistical and analytical parameters, such as e-value and p-value, are induced to describe the significance of the motifs.

E-value (Expected value)

In probability, the e-value of a random variable is the sum of the probability of each possible outcome of the experiment multiplied by its payoff ("value"). Thus, it describes the likelihood that a motif with a similar score will occur in the sequences by chance. The smaller the e-value, the more significant the alignment appears with the group of patterns relative to the background set.

If X , motif, is a discrete random variable with N values x_1, x_2, \dots and corresponding probabilities p_1, p_2, \dots which add up to 1, then $E(X)$, expected motif appearance possibility in the background sequence, can be computed as the sum or series

$$E(X) = \sum_{i=1}^N p_i x_i \quad (14)$$

The e-value functions to filter out those motifs beyond the threshold, which have high score (frequency) in the background, because we assume the motifs should have significant score in the target sequences instead of the background.

P-value

In statistics, p -value is the probability that an associated null hypothesis is true given a particular set of observations [25]. Typically, this is the probability that a particular set of observations can be explained entirely by chance. A cut-off is normally set below which the p -value indicates that the null hypothesis is false and it implies that the observations cannot be explained by chance alone.

Assume the background sequences have N annotated ones of which K have the specific classification of interest (e.g. possess the same motif). Then the probability that a randomly selected background sequence has that classification is $p = K/N$. If a particular

cluster has n classified sequences, of which k have the classification of interest, it is important to determine the probability of observing k or more random events of probability p from a set of n . Thus, intuitively, the p -value may be computed from the Hyper-geometric Distribution [26]:

$$p = \sum_{i=k}^N \frac{\binom{k}{n} \binom{K}{N-n}}{\binom{k+K}{N}} \quad (15)$$

If a particular cluster and classification combination pass the p -value criterion, this indicates that it is accepted statistically the number of observed occurrences of the classification in the cluster cannot be explained by chance, i.e. the cluster is statistically biased towards the classification.

A significantly smaller p -value criterion would be required for sequences clusters based on less reliable data, such as expected. Because the relationship between sequence and function is so well established empirically, the null hypothesis is implicitly false, and less statistical evidence is required to establish selective bias than with other clustered data.

Experimentally, the value of 0.01 was found by manual examination of borderline classification outcomes on a large test dataset; values for other types of clustered data can be determined by similar mechanisms. It is suggested that different p -value criteria be used depending on the reliability of the clustered data.

3.3 Overall program flow-chart

The program flow chart that implements any of these two algorithms is depicted in Fig 3.3. Program flow blocks consist of two main portions: instruction and decision block.

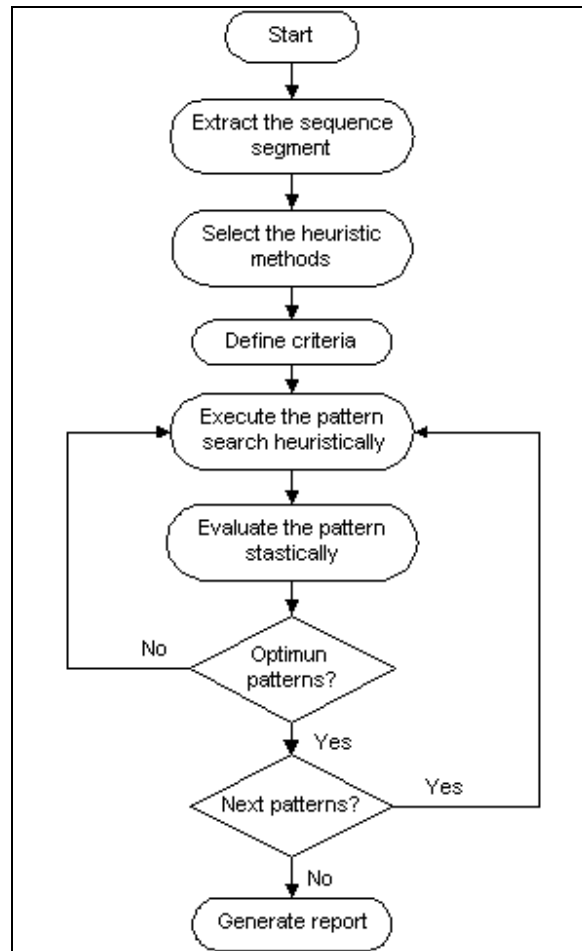


Figure 3.3 Main program flow-chart

Instruction block: it is the procedure for the software to operate and manipulate the data

- **Extract the sequence segment:** to specify and obtain the sequence segment, which the user is interested in.

- **Select the heuristic methods:** to choose the suitable algorithm to extract the homogenous motifs from the sequences' segment
- **Define criteria:** to set the parameters for the heuristic algorithm, for example, threshold for the EM, expected motif length, and etc. The parameters are very important for the system to predict the motif group accurately.
- **Execute the pattern search heuristically:** to run the heuristic search on the patterns by following the criteria for the algorithm.
- **Evaluate the pattern statically:** to compute e-value and p-value in the background sequence by statistically approach, such as HMM
- **Generate Report:** to generate the graphical and text format report to describe the pattern groups and their parameters.

Decision block: it is the defined criteria for the program to execute the instruction

- **Optimum Pattern:** to evaluate the pattern whether is optimal among several searches by comparing the statistical score and the fitness among the population. It stops when it reach optimal point, otherwise, it continues searching.
- **Next pattern:** to check whether the next pattern search is still required. It will continue searching when the number of patters is incomplete.

Chapter 4 Transcription Start Site Viewer (TSSViewer)

The long-term objective of gene regulation is to enhance our understanding of transcription process by elucidating its key components and their functional relationships. Bioinformatics analysis of promoters can significantly contribute to this goal. However, the promoters contain a large number of elements (PE) that appear in various combinations of different promoters. Moreover, currently no PE common to all promoters are yet found. So, the promoter structure is characteristic for a smaller gene groups, likely those that are co-regulated. Transcriptionally co-regulated genes are those whose transcription is controlled by very similar set of TFs. Consequently, such genes can more frequently co-express together. But since there are numerous PEs that can be detected computationally, it is a headache for biologist to analyze such data for a large number of promoters. Therefore, the system for visualization of promoter content and promoter structures is of a great practical utility. We developed one such supporting system that is implemented in a database of human promoters as a valuable tool for biologists to analyze and interpret the complex and huge volume of promoter data.

4.1 Problem Statement

Presentation of PEs near TSS in a strand is one of complicated problems in the bioinformatics study, because PEs could appear in various combinations. Thus, a lot of information is essential to describe PEs in a way that it may be useful for biological interpretation. This information includes the actual PE pattern, the combination of PEs, motif location, over-representation relative to the background sequences, etc. Although

such information could be provided in the text format database, it is cumbersome to address the problem of simple inspection of promoter content in a systematic manner. For example, one PEs (or combination of PEs) with close (or overlapping) location to another PE is not easy to express and observe in tabular form, while it is simple to do it through visual presentation. Therefore, a suitable, comprehensive and systematic presentation approach is essential for biologists to analyze and interpret PEs, their positional distribution and their associated information.

Thus we define the problem we intend to solve:

1. Design a suitable method to present PEs, their positional arrangements, and their associated key information for graphic application in a promoter database. Enable interactive information reading.
2. Develop a system to automate generation of graphic files suitable for integration in a promoter database.

4.2 Objectives

Realizing the difficulties in the biological information, it is recognized that a graphical representation is essential to describe the complicated composition of PE and their positional arrangement in the initial phase of project. Specifically, it is suitable to have ability to present the content of promoters and its organizational features expressed in terms of PEs, combinations of PEs, and their distributions, so that such compositions can be analyzed visually.

4.3 System Description

TSSViewer is the system for visual presentation of information about regulatory patterns found computationally in the promoter regions of different genes. The graphical presentation provided is flexible and portable and enables an effective inspection of promoter database for detailed analysis of promoter properties. This system describes the relationship between the positions and structure of PEs in the selected promoter indicating also their relation to the Transcription Starting Site (TSS). Such information is essential and fundamental for analyzing the causes of the gene expression, and in determining the importance of PE and their relation to gene functions.

TSSViewer is developed as a perl program. It is integrated into the Dragon Regulome (Hs) Database (Dragon REGHSdb) (<http://research.i2r.a-star.edu.sg/DRAGON/REGHsdb/>). Dragon REGHSdb is the first database of the Dragon suite of tools and databases, which focus on the transcriptional regulatory motifs in the promoter region covering [-250, +50] positions relative to TSS. This database includes information about 1800 promoters of human genes. All TSSs are collected from the Eukaryotic Promoter Database (EPD) [53]. In order to view the promoter content it was necessary to develop a graphical interface for Dragon REGHSdb, so that promoter structure of promoters in the database is available for inspection. The graphical images, which describe promoter contents, were generated by TSSViewer and they also contain information on the type of TFBS, its location, and strand. The graphical interface built in the database allows for the interactive work with the graphic promoter content files.

4.4 Software Description

To generate files that contain graphical representation of promoter content, we developed TSSViewer program. This program is mainly used in the Linux and Unix operating systems. The software is developed in Perl and uses the GD library [48], one of the graphical tools for image creation. Among other formats, the GD library could generate PNG, JPEG and GIF images. TSSViewer system used this library to convert text database description into the image formats.

Besides the images generation, TSSViewer software also includes the several other functionalities, such as data acquisition and generation of html files. Data acquisition is necessary to obtain and classify the TFBSs into different groups according to their associated information, such as position and strand, etc. Moreover, the html file, associated with the image, is used to label TF information using the Javascript technique. This Javascript method enables that some attributes of information are presented in the pop-up window, which makes the presentation more interactive.

4.5 File Format

PEs in the Dragon REGHSdb, which are the input data for the TSSViewer system, are obtained by filtering predictions produced by the Match program [54] of Biobase, Germany, and are based on the TRANSFAC database (public version 6.0) [55].

The input database consists of the TFBSs, which are mapped to human promoter sequences obtained from the Eukaryotic Promoter Database (EPD). The input file is presented in a text format.

Sample dataset input text

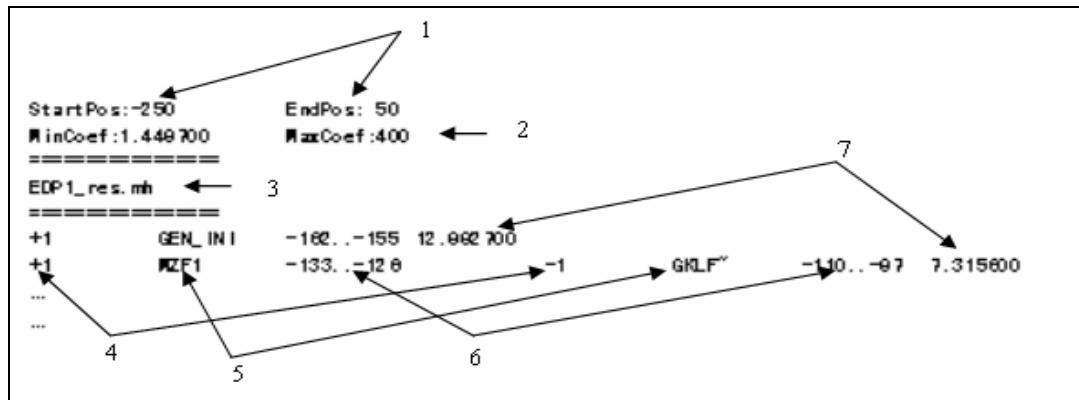


Figure 4.1 Snapshots of the TFBSs description entry

The input data contains different information:

1. The segment start (StartPos = -250) indicating where the promoter region considered starts (that is 250 nt before the TSS); the ending position of the segment (EndPos = 50) that indicate that the analyzed segment ends 50 nt after TSS.
2. The range of the over-representation index (ORI) [57] of PE relative to the background data. In the Fig.4.1, this information indicates that ORI ranges from 1.449 to 400.

If the PE is made up of single TFBS then:

3. The name of the file that contains initial results of mapping of PE to promoter sequence.

4. It indicates the strand where TFBS is found (+1 or -1 stand for positive or complementary strand, respectively).
5. The name of TFBS pattern found is shown.
6. It provides the actual location of the TFBS expressed relatively to the known TSS.
7. Also, it shows ORI for the specific PE.

If the PE is made up of two TFBSs that are detected within certain prespecified distance (in the case of Dragon REGHSdb this distance was maximum 50 nt), then information is given first for the one TFBS followed by the other TFBS. The ORI value is given for the pair of such elements and it is given as the last number in the row (Fig 4.1).

Visualization dataset output is presented as a form of html file, which is linked to the image file that describes the TFBS information.

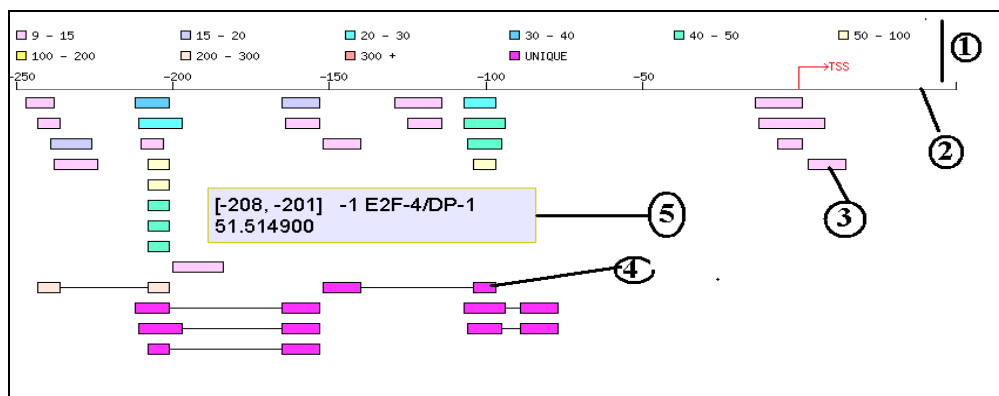


Figure 4.2 Snapshots of the output file

Generally, the image representation as given by an example if Fig. 4.2, can be divided into five portions,

1. Top layer. This portion contains the labels for indication of the range of ORI based on color. Several square boxes are presented each colored differently. Next to the box the actual numerical range of ORI is given. These colors are used to color individual PE. For example: \square 15 - 20 means that all PEs represented with this color are 15 to 20 times over represented as compared to the background sequences. The term UNIQUE means that PE with that color was not found in the background sequences, but only in the promoters.
2. The line symbolically represents DNA segment. The numbers on the line indicate the position relative to TSS location, that is, the number of nucleotides upstream or downstream as shown below:



TSS: arrow indicates the direction of the gene.

-50: 50 nucleotides upstream of TSS.

50: 50 nucleotides downstream of TSS.

3. The rectangular boxes indicate single PE. These are the TFBS which are mapped to the positive or negative strand. Their color indicates the range of ORI for that PE. Their length approximately corresponds to the actual PE length expressed relatively to the length of promoter segment analyzed.
4. When pair of PEs is found, they are presented as a pair of rectangular boxes linked with a straight line. Their color indicates the range of ORI for that pair of PEs.

5. When the mouse cursor is positioned on the PE that is on the graphical presentation of promoters, it activates the pop up block that displays the associated PE and pair of PEs in more details. Examples of how these displayed information blocks may look are given below. The pop up windows contain the actual positions of TFBS given in square brackets, TFBS strand, TFBS name and ORI. If it describes a pair of PEs, then it contains information for individual PE, as well as ORI for the pair.

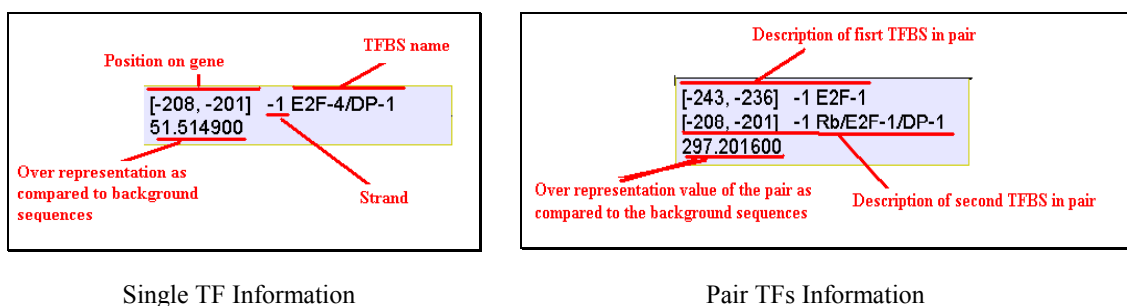


Figure 4.3 The content of pop up windows

4.5 Program Flow

In this section we present and describe the flow chart of TSSViewer program. The flow chart is depicted in Figure 5.4. The block diagram consists of the file information and the instructions to manipulate the files.

File Information: These are the files manipulating in the program. It composes of the input and output files.

- **Input File:** This is the data file, which contains the TFBS as shown in Fig 4.1.
- **Images / HTML file:** These are the output files generated for the program. the image file, which contains the information for the PEs near the TSS as shown in

the Fig 4.2, and the HTML file, which contains the information for the each PEs with the pop-up window in the Javascript, as shown on the 4.3.

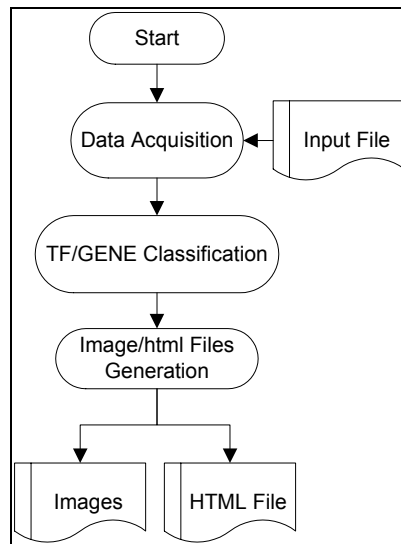


Figure 4.4 TSSViewer Program Flow Chart

Instruction: The procedures in the program, which is to acquire data, classify the information and generate the graphic output.

- **Data Acquisition:** to acquire the PEs / genes information from the input files.
- **TF/Gene Classification:** to classify PEs / gene based on their transcriptional relation shown in the input file.
- **Image/html Files Generation:** to visualize and present the relation between PEs and the associated genes in format of graph and html files, with the assistant of the GD library.

4.6 Comment on TSSViewer

As discussed and illustrated in the previous sections, it was found that the graphical presentation of promoter content provides a valuable utility for biologists as they can analyze positional distribution of motifs by which promoters are annotated. Moreover, one can inspect the type of motif by moving the mouse over that element block. Even more, combinations of two PE that are found within the maximal mutual distance of 50 nt found in promoters could also be inspected.

Such insights are not possible through tabular representation of data. Consequently, the database with such visual representation of promoter content enables different means for biologist to get insight into promoter structure of his target gene groups.

Chapter 5 MotifBuilder and the web application

Besides the heuristic algorithms to obtain the motifs, in this study we developed software for interactive visual presentations of biological data in MotifBuilder system. It is also used as a part of input for the graphical representation of transcription regulation networks described in Chapter 6.

5.1 Problem Description

The heuristic models in Chapter 3 have been introduced as the tool to extract families of mutually very similar motifs from the sequences of interest. Information about these motifs could be vital for biologists to distinguish them from the spurious DNA patterns contained in the sequences. If the analyzed sequences are promoters, then the extracted motif families have high likelihood to correspond to potential TFBSs. Thus, it is essential to describe these potential TFBSs information not only as a family but also as individual patterns. Also, there is an issue of arrangements of such elements when analysis of co-regulated or promoters of orthologous sequences are analyzed. Even though the tabular presentation could provide the exhaustive information, it is not easy and comprehensive for the user to visually inspect and scan the statistic measures associated with motifs so as to identify the potential TFBSs. Thus, the graphic presentation is necessary to complement the tabular one in description of the motif distribution along the sequences.

5.2 Objectives

The system we name MotifBuilder, was developed to provide reports in tabular and graphic form to present motif information. Compared to the exhaustive tabular presentation, the visualization reports are convenient to represent specific and complex information of the potentially important biological patterns found in multiple sequences (such as putative TFBSs, their cumulative distribution and distribution along individual sequences). This way, we may inspect for example, the preservation of arrangement of motifs found in a set of sequences.

5.3 MotifBuilder Description

Dragon MotifBuilder (DMB) (http://research.i2r.a-star.edu.sg/DRAGON/Motif_Search/) is the analytical system for determining sets of homogeneous patterns from a set of unaligned or aligned sequences and for graphical presentation of the found motifs. The system is developed with the C and Perl languages, and is compatible with different operating systems, such as Unix, Linux and Windows.

DMB consists of two main portions. One is the heuristics based computation and data extraction, while the other is the data summary report. The first portion, heuristic computation, aims to extract the pattern information with algorithms which have been developed and described in Chapter 3. The summary report is the portion that describes and represents the pattern information and their cumulative distribution, as well as distributions across the analyzed sequences. The motif report consists of two types of reports: tabular and graphical.

5.4 Motif Report

The motif data in text form focus on the expression on the individual patterns appearance in the sequences. Therefore, the motifs are identified as a group which has a high pattern similarity, and the actual motif patterns are presented and described in the report too.

Motif report produces two html files that contain different text format information. One of the reports for the motif group patterns is shown as the following Figure 5.1, which aims to provide the individual pattern information.

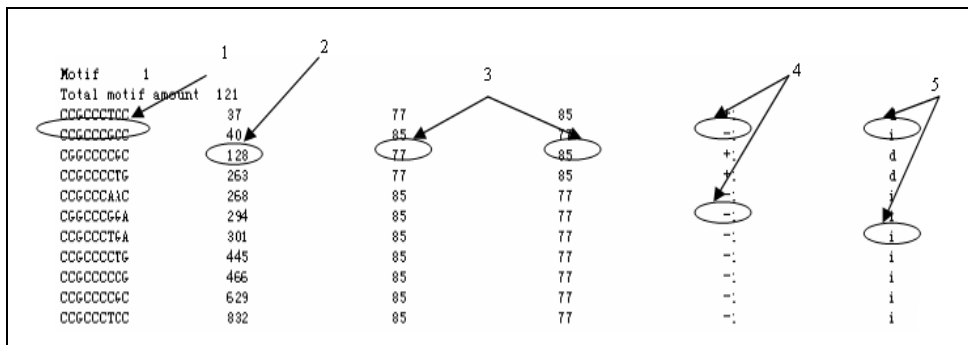


Figure5.1 Motif report from the heuristically search

The explanation of the annotation in reports page from Figure 5.1 is as follows:

1. Denotes a specific motif pattern which belongs to the conserved motif group
2. Denotes a specific sequence in which the motif pattern is found
3. The start and end position of the motif in the sequence
4. The strand of the DNA sequence where the motif is found, +1 → forward strand; -1 → complementary strand.
5. The sequence orientation, d → direct orientation; i → inverse orientation

The other file presents the summary report form in term of PWM for the motif family. The other relevant information for the motif family is also presented such as statistical measures, p-value, e-value and information content for the group of motifs.

```

=====
Number of sequences used for PWM is 64 (50.39%)
CONSENSUS PATTERN: CTATAAA
Threshold for PWM score is 0.900
e-value for PWM is 0e+000 (tested on 1000000 nt) with Threshold 1e+000
p-value for PWM is 1e-040 for k = 64, n = 127, K = 304, N = 4126 with Threshold 1e-002
Population -> position mean 939 and sigma 41
=====

```

A	C	G	T	InfoCont	Position	Pattern
17	24	12	11	0.071864	1	C
0	0	0	64	2.000000	2	T
64	0	0	0	2.000000	3	A
0	0	0	64	2.000000	4	T
64	0	0	0	2.000000	5	A
64	0	0	0	2.000000	6	A
64	0	0	0	2.000000	7	A

```

=====
TOTAL INFORMATION CONTENT          12.071864
=====

```

Figure 5.2 Tabular representation of the PWM for a motif family in the html file

Figure 5.2 presents the total number of motifs found as belonging to this motif family, and the percentage occurrence relative to the total number of sequences. Second row describes the consensus pattern of such motif family. For example, CTATAAA, is the consensus pattern obtained for the motif group as a whole. The selected threshold coefficient for the algorithm to extract the motif group is also shown. Additionally, we present some statistical measures such as e-value and p-value, which are used to describe the over-presentation of the motif family in the target sequences as opposed to the background sequences. PWM is constructed to express the similarity and consensus of the motif group. The consensus nucleotides for each position are given, sometimes indicating alternative nucleotides (the most abundant bases). The information content for

each of the positions for the PWM, as well as the information content for the overall family is presented.

5.5 Visual Presentation of Motif Information

All graphical information for a motif family is classified and presented into two catalogues, according to the cumulative motif group position distribution and the distributions of individual motifs across all sequences.

For the individual motif population, the positions for the group of patterns are identified and summarized as the distribution list in Figure 5.4. The percentage of the specific position bins are annotated on the diagram.

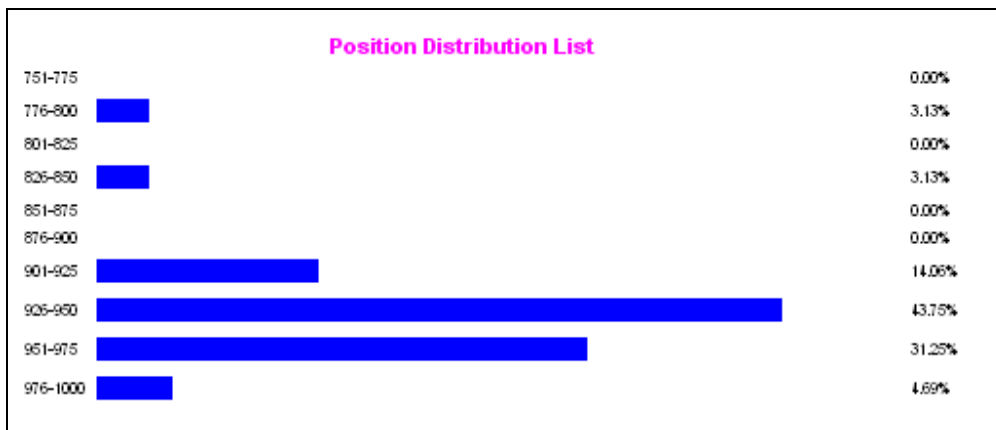


Figure 5.3 Starting position distribution list for one group of motifs

This position distribution chart illustrates the rough position distribution of members of the motif family and in some cases makes it possible to identify motifs that show high bias in the positional distribution. This is of particular relevance in the case

when the original sequences are aligned because then the positional bias is an unexpected event, and likely could be related to biological significance. For example, the well known cases of TFBSs that show strong positional bias are TATA box, downstream promoter element, GC box, Sp1 [57].

On the left side of the graph we present the information about positional bins, such as (751-775) that indicates the segment of the sequences, where the motif appears. The data on the right side indicates the percentage of the motifs that occurs in the specific positional bin. The center bar chart visualizes the percentage according to the data at right.

```

<TABLE BORDER=0 height="5">
<TR><TD COLSPAN=2 height="5"><p align="center"><font size="2" face="Arial" color="#FF00FF"><b>Position
Distribution List</font></TD></TR>
<TR><TD height="5"><font size="1.8" face="Arial">751-775</font></TD>
<TD><TABLE>
<td width="48.979592" height="5" bgColor="blue" align="center"><font size="1.8" face="Arial"
color="white">&nbsp;</font></td>
<td width="93.877551" height="5" bgColor="white"></td>
</TABLE> </TD>
<TD height="5"><font size="1.8" face="Arial">6.12%</font></TD>
<TR><TD height="5"><font size="1.8" face="Arial">776-800</font></TD>
<TD><TABLE>
<td width="97.959184" height="5" bgColor="blue" align="center"><font size="1.8" face="Arial"
color="white">&nbsp;</font></td>
<td width="87.755102" height="5" bgColor="white"></td>
</TABLE> </TD>
<TD height="5"><font size="1.8" face="Arial">12.24%</font></TD>
<TR><TD height="5"><font size="1.8" face="Arial">801-825</font></TD>
<TD><TABLE>

```

Figure 5.4 HTML expression format for the position distribution chart

The position distribution chart is produced as a simple HTML file. Although it is not convenient for precise graphical presentation it provides a fast and effective solution to express the complex problem simply. The graph is constructed with the table form in HTML, as shown in the example in Figure 5.4.


The other motif distribution diagram in MotifBuilder is related to representation of the distribution of motifs in the set of sequences from which motifs are identified. If motifs are selected from a group of promoters that are related and whose sequences are aligned relative to TSS, then we would expect to observe in many cases some preservation of the promoter content between the promoters. This could be reflected as the preservation of the distribution of some of the motifs and preservation of their mutual distances. But the only convenient way to observe such preservation is through the visual representation of promoter context. So, this motif distribution diagram provides the overall foothold of the potential TFBSs and their locations in promoters, and it could help biologists to inspect, analyze and discover the actual biologically relevant motifs with their position correlation.



Figure 5.5 Motif distribution in the promoter region [-250,-1] relative to TSS, for mouse H4 histone gene group.

Figure 5.5 represents the positional distribution of motifs identified by MotifBuilder in a set of 127 sequences of mammalian species (man, mouse and rat). The regions covered the range of [-250,-1] relative to the TSS location [56]. They contain 127 histone gene sequences with 19 H1, 29 H2A, 32 H2B, 23 H3 and 24 H4 histone type. We found that the five mammalian histone gene groups (H1, H2A, H2B, H3 and H4) have

mutually distinct, prominent and strongly conserved regions with motif modules in the upstream region of the TSS. Moreover, they are also reasonably well conserved across the same species. In the Fig 5.5, the sequences are from mouse H4 histone gene. These sequences show strong similarity in terms of motifs and their positional distribution. The motifs identified correspond to the known TFBSs. For example, motif 1 identifies the CCAAT box, and motif 3 identifies TATA box. What is important for us to notice is that identified motifs show strong preservation of positions relative to TSS across promoter sequences. This is one potential indicator that motifs do not appear randomly distributed and thus suggest that they may be biologically active, which is true in our case.

In the motifs distribution diagram along the sequences as shown in Figure 5.5, the motif are presented in the corresponding sequences proportionally to their location relative to the sequence length. For explanation,  means the third reported motif and “+” indicates that the motif is found on the forward strand (“-” means complementary strand), while “d” indicates that the pattern appear in the direct orientation, while “i” indicates that the pattern appears as inverted sequence. Additionally, different colors associated with the label help to more easily distinguish different motif groups, particularly when we present a large number of motif groups. Motif distribution diagram is also constructed as the HTML table.

Besides the two different distribution diagrams, another Perl program, MBCConvert, could translate the text form MotifBuilder report into the input format of TFMapper, which another system for graphical representation of transcription regulation information that will be explained in Chapter 6. Therefore, the interconnection network

of the motifs and their sequences could be presented with the TFMapper system as illustrated in Fig 5.6.

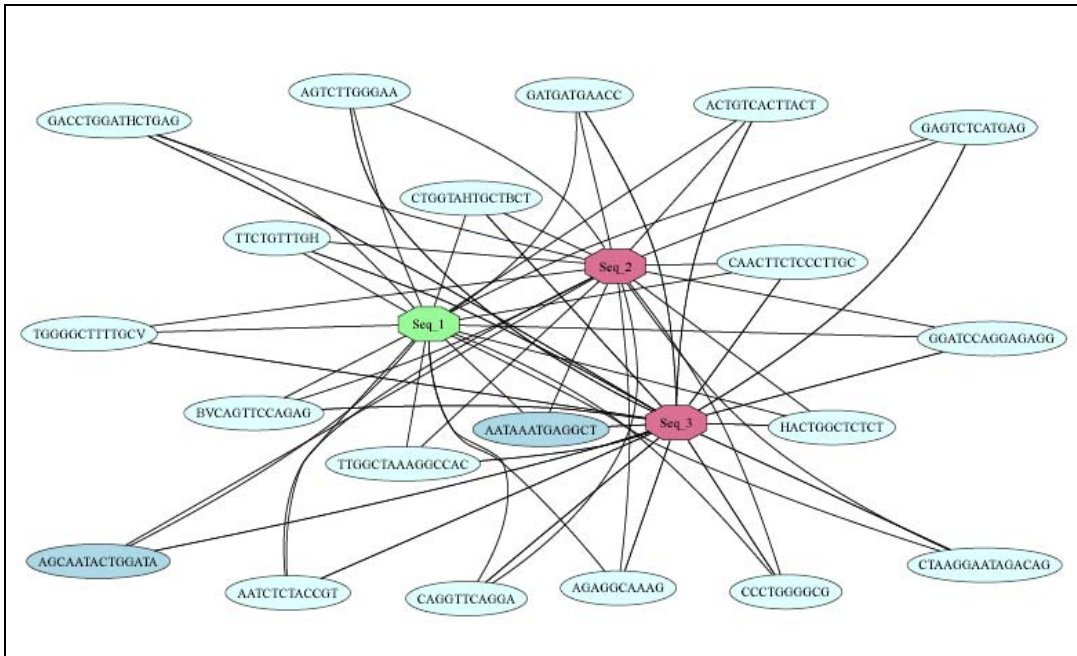


Figure 5.6 Interconnection Network between the motifs and sequences

5.5 Visual Presentation of Motifs

In the DMB system, one of the modules caters for visual presentation of motifs that are identified. The flow chart of the section of this module that generates reports (which contain graphic presentation of motifs) is depicted in Figure 5.7. The blocks in the flow chart are described below:

Data block consists of the **input** and **output** data reports, which present the motif information

- **Input data obtained from “Heuristic Search”:** This block is used to find out the homogenous motif with the guide of the heuristic algorithms, as discussed in

Chapter 3. The intermediate results of produced by this activity serve as input to report generation module.

- **Output file “HTML file” and Image File:** These are two different formats in presenting the motifs information, the HTML is for the tabular one, as described in Fig 5.1; the image format is for the distribution diagram, as shown in Fig 5.3 and 5.5.

Main block consists of the procedure blocks to generate the text report and visualize it.

- **Generate the text report:** This block is used to collect all the motifs found out by the heuristic search, and present them in the html format, as shown in Fig 5.1 and 5.2. All these text reports could be directly viewed with the internet browser.
- **Translate into image file:** This block is used to visualize the text format file, and generate the graph presentation for these text files, as shown in the Fig 5.3 and 5.5.

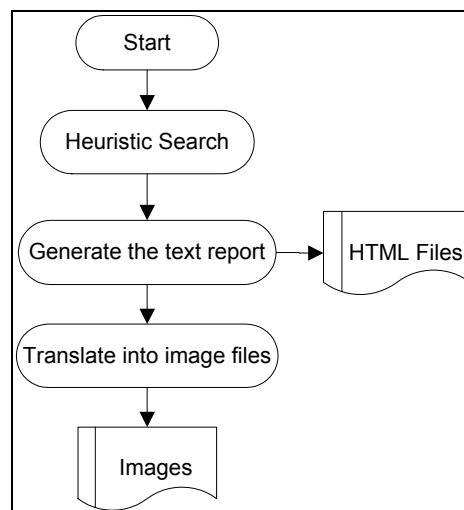


Figure 5.7 Schematic presentation of the module for generation of reports that contain graphics.

5.6 Web-based Application

Dragon Motif Search Tool (DMST), for extracting and presenting sets of compact patterns from a set of unaligned sequences has been developed with heuristic algorithm in format of web-based application. This application is integrated with four different heuristic methods for motif clustering. Besides the two algorithms which are introduced in Chapter 3, two other methods, such as 'tabu' search [28, 29, 30] and simulated annealing [31, 32, 33, 34, 35], are also implemented. The algorithms share certain similarities with genetic algorithm approach, but do not operate on generated populations of patterns.

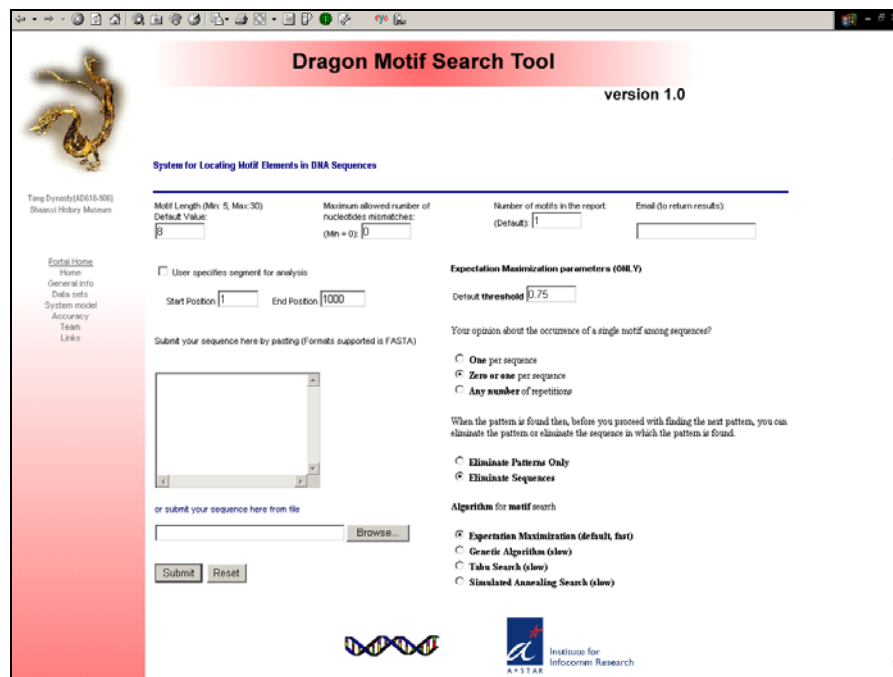


Figure 5.8 Snapshot of the Dragon motif search tool

This web-based tool can be directly applied in determination of potentially functional patterns in DNA. The system is available as a public web application free for

academic and non-profit users and can be found at http://sdmc.i2r.a-star.edu.sg/DRAGON/Motif_Search/.

5.6.1 Dragon Motif Search Tool

The DMST aims to provide a free-access tool for the biologists to analyze the biological sequences. Therefore, the web-browser is used to acquire the sequences and pass it to server for the analysis, and then the report would be mailed to the users.

5.6.2 Procedures and Operations of Dragon Motif Search Tool

The main page of the tool is shown in Fig 5.8. The operation of the tool could be divided into the following procedure.

a) Input File preparation.

In order to use this tool, users should provide a set of aligned or unaligned sequences in the FASTA format [36]. These sequences can be either pasted to the main sub-window provided, or the ASCII file in the user's computer, which contains FASTA sequences, can be browsed through the smaller sub-window below the main one by using the 'browse' key. After pressing the 'submit' key the file or pasted sequences will be transmitted to the server and further processed.

b) User Email Information

Due to the long consuming time to run the heuristic search, users should provide their e-mail address, so that they could receive the report of searching results. Without this email information, the system will not produce any output.

c) The other options provided for all implemented methods include:

c.1 motif length; the default is 8 nucleotides; ranging from 4 to 30 nucleotides.

c.2 number of motifs (motif groups) in the report; the default is one;

c.3 if the sequences are aligned, then it is possible to select the segment for submitted sequences to be analyzed; for this users need to check the square box before the 'User specifies segment for analysis' and then select the start and end positions of sequences for the analysis.

c.4 an option to either eliminate a sequence if it contains a pattern which will be included in a group, or to mask by 'N's such a pattern; to select these options users have to use 'radio' buttons; the default is 'eliminate sequence'.

c.5 the checkbox to induce the double-stranded search for all the algorithms.

d. Specific algorithm selection

d.1. EM

The default method in the DMST is 'Expectation Maximization' algorithm since it is efficient to obtain the analysis results. In the EM-based algorithm the pattern will be included in the group if its matching with the position weight matrix (PWM) generated from the previously selected patterns is above the selected threshold. In this case users additionally can select:

d.1.1. the threshold, which ranges from 0 to 1; the default value is 0.75.

d.1.2. average Information Content threshold, which ranges from 0 to 2; the default value is 0.85

d.2. Genetic algorithm, Tabu Search and Simulated Annealing

d.2.1. select the maximum number of nucleotide mismatches allowed for a new pattern to be included in a group;

d.2.2. use option (by means of radio-buttons) to select the mode of operations of these algorithms so as to allow that exactly one pattern be selected from each sequence during iterations, or that maximally one pattern (the best) from a sequence be included in the group if it satisfies the required conditions, or that any number of patterns from a sequence could be included in the group if they satisfy the required conditions.

5.7 Other Applications

Additionally, the report pages of DMST have been integrated into the Dragon Explorer of Estrogen Responsive Gene Functionality (DEERGF http://research.i2r.a-star.edu.sg/DRAGON/FERGDB1_0/).

In the example given below in Figure 5.9 a, DMST has identified motifs in three ortholog promoter sequences for gene ATF3 (activating transcription factor 3, which represses transcription from promoters with ATF binding elements [58]) from human, mouse and rat. A typical question of interest to biologist is what are motifs that are common between the members of the ortholog group and are the positional organizations of motifs preserved. It can be observed that the block of four motifs remain conserved between human and mouse (Figure 5.9b), showing that the motifs are also preserved relative to their positional. The blank parts of the promoter sequence represent positions where other identified motifs are located but these have not been found common between the two promoters. Thus biologists have a clear picture about what are common motifs in

these ortholog sequences, but also have an idea about distribution of other motifs. Moreover, when we look for the common motifs between human and rat, we find that they share one motif more (motif 13) which does not appear in mouse ortholog.

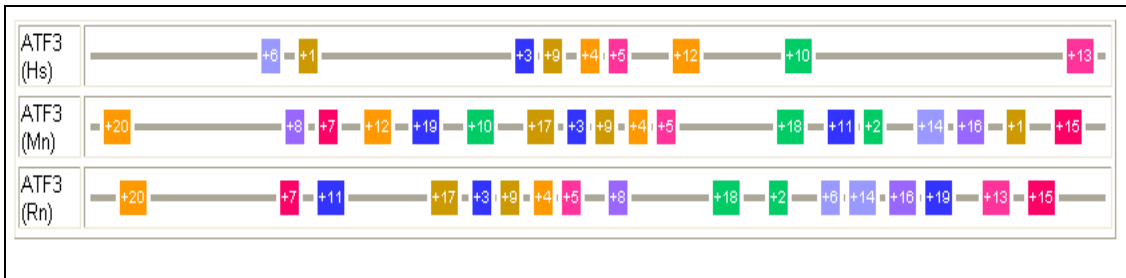


Figure 5.9a

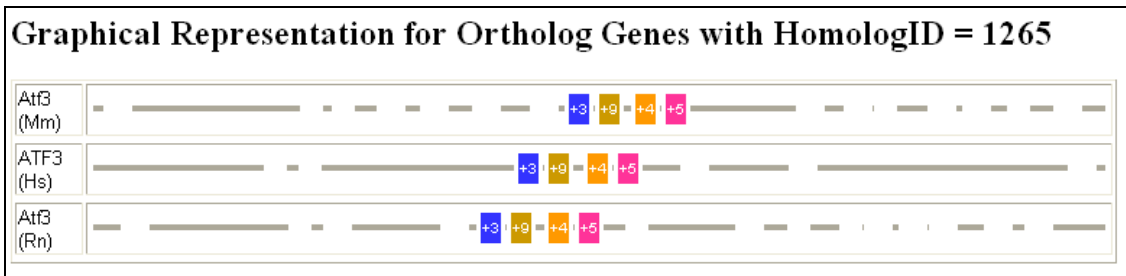


Figure 5.9b

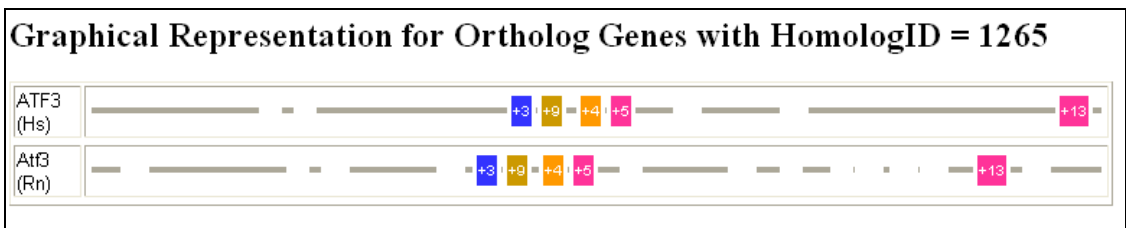


Figure 5.9c

Figure 5.9. Snapshot of the promoter content of ATF3 ortholog genes. In the case of human and rat (5.9c) there are more common promoter elements that have preserved positional organization, than is the case when human, mouse and rat are considered (5.9b). This suggests mouse specific solution in promoter composition for the ATF3 gene.

This example illustrates how useful systems of this type could be and how graphical representation makes convenient medium for biologist to get insight into promoter features.

Chapter 6 TFMapper

With the increased interest in understanding biological networks, such as protein-protein interaction networks and gene regulatory networks, methods for generating such networks and then representing them becomes increasingly important. It is our interests to develop tools which could generate the network map out of genes and the PEs that potentially control them, so as to obtain a putative transcriptional regulatory network. However, this transcriptional regulatory network is not difficult to unravel and present because the complex relation between the TFBSs and the associated genes. Thus, one simple but effective network program has been developed in our study. This section presents such development that generates transcriptional regulatory networks suitable for analysis of role of PEs in control of various genes.

6.1 Objectives of the Development

It is our objective to develop TFMapper system that aims to assist biologists to reconstruct parts of transcriptional regulatory network. This system utilizes the promoter content based on the input data files produced by other systems that map important PEs to the promoter. Then, TFMapper will be designed to analyze this information, extract the interconnection between the PEs and genes, and provide the graphic layout to illustrate the relationship between the genes and TFs. The system needs be developed as a Windows application. It aims to be an effective and efficient solution to suggest the correlation of genes and TFs.

6.2 Software Description

With the clear design objective, TFMapper was developed as a novel graphic program, which makes use of the Graphviz [37] package for drawing associated graphs. TFMapper is a stand-alone software with the graphical user interface. It, however, requires input data files in specific format (as described in Chapter 5). Then it uses these files to generate layout of the graphic network. Basically, the software consists of modules for:

- acquiring data,
- manipulation of data,
- graphical layout generation
- graphical user interface (GUI).

The system is developed in Visual C++ 6.0, which is compatible with the Windows operating system. Compared with the previously described two graphic presentation programs (in Chapters 4 and 5), this system emphasizes more on presenting information that makes connection between the genes.

The GUI of the TFMapper is shown in Figure 6.1. This GUI generally is composed of five different portions that deal with various types of information required or generated by the system:

1. File Information:

This part collects information about the input and output files and their directories.

2. Number of TFBSs shared by the genes:

This is information that users supply. For the genes that will later be selected by the user, the link between them will be characterized by at least the number of TFBS motifs shared by the promoters of these genes. These will be displayed on the layout.

3. List of name box for the TFBSs and genes, and user specified TFBSs and genes:

Double clicking the specified TFBS list box displays the number of gene with selected TFBSs.

4. Number of genes with the selected TFBSs:

Double click the TFBS checkbox and the number of genes with selected TFs will be shown in the textbox.

5. Utility function keys:

There are several keys provided to help user in specific tasks. These are

- a. Get TFBS: to extract all TFBSs patterns from the input data.
- b. Select/ Remove: to choose or delete the highlighted TFBS/gene.
- c. Generate: to generate the image with the information.
- d. Reset: to reset all the information in the list box.
- e. Refresh Gene: to update the corresponding gene information with the known TFBSs.
- f. View: to view the images generated by the program.

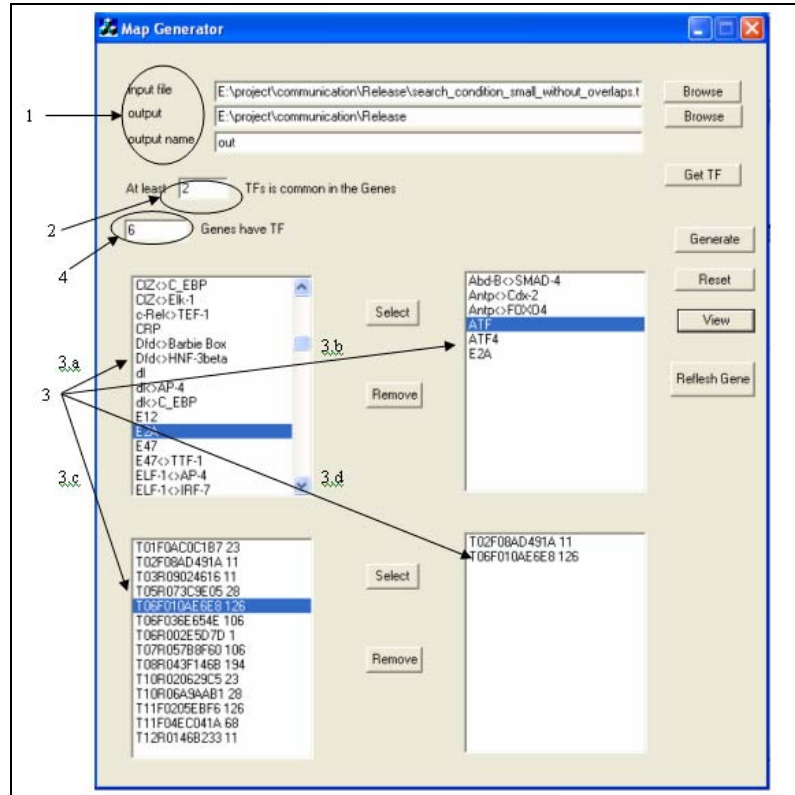


Figure 6.1 Graphical user interface of the TFM Mapper

6.3 Working Principle

Users need to provide one input data file which contains information about TFBSs and genes they are controlling. User has a possibility to select TFBSs he/she is interesting investigating, and all genes that are putatively controlled by more than a specified number of the selected TFBSs, will be extracted. Certain non-obvious TFBSs relationship between the extracted genes and TFBSs that have not been selected will also become available and presented in graphic layout. For example, TFBSs associated with the user specified genes will be shown in the graph, and interconnection relationship of these TFBSs and other genes will also be shown.

6.4 Using TFMapper software

Below we provide list of instructions how to use TFMapper.

- Click the input file Browse button to identify the input file path.
- Click the output directory Browse button to specify the output file directory.
- Fill in the output file name without any extension in the output name textbox, because the output file extension is defined svg format in the system.
- Click the Get TF button to extract all the TFs from the input file and list on the left-top list box.
- Highlight the specific TF in left-top list box 3.a, and press the Select button to choose the TF to present in the relationship map. Then the TF will be put into the right top list box 3.b.
- Press the Refresh Gene and the genes with the selected TFs in the list box 3.b will be automatically extracted and listed on the left bottom list box 3.c.
- Highlight the selected TF in the right-top list box 3.b, and press the Remove key to remove the TF if user would not like to choose the TF presenting in the map.
- Double click the selected TF in the list box 3.b, the number of genes with the selected TF will be displayed.
- Highlight the select gene and press Select key if more detail need be shown in the map for specific gene in the left-bottom list box 3.c. Then the specific gene will present on the right bottom list box 3.d.
- Press the Generate button to generate the image according to information genes and TFs that user define.
- Press the Reset button to reset all the information in the list-box.

- Press the View button to view the image.

6.5 Input / Output File Information

The input data shares the same format as the one in the TSSViewer, as shown in Fig 4.1 as in Chapter 4. TFMapper classifies the TFBSs according to the genes they are putatively controlling and translate them into the format of the input file for Graphviz as in Fig 6.2.

```
graph GeneMap{
"T04F076A190E 156"[shape=octagon,fillcolor=palevioletred,style=filled];
"T11R04F46CCB 106"[shape=octagon,fillcolor = palegreen,style=filled];
"T15F03E4A886 194"[shape=octagon,fillcolor = palegreen,style=filled];
"Hb"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "Hb";
"TTF1"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "TTF1";
"E47"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "E47";
"E12"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "E12";
"PITX2"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "PITX2";
"AML-1a"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "AML-1a";
"AML1"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "AML1";
"Osf2"[fillcolor=lightcyan,style=filled];
"T04F076A190E 156" -- "Osf2";
"TCF-4"[fillcolor=lightcyan,style=filled];
```

Figure 6.2 Translated Input file for Graphviz.

The input file for Graphviz contains the relation of TFBSs and genes. When user specifies genes and TFBSs of interest, the system will extract the relevant information for relationship network reconstruction. In the network the specific shapes and colors are assigned to TFBS and to genes.

Once the input file for Graphviz has been generated, the TFMapper system will execute the Graphviz to generate the network image. In the network, the text term description for the features will be decoded as illustrated in Figure 6.3.

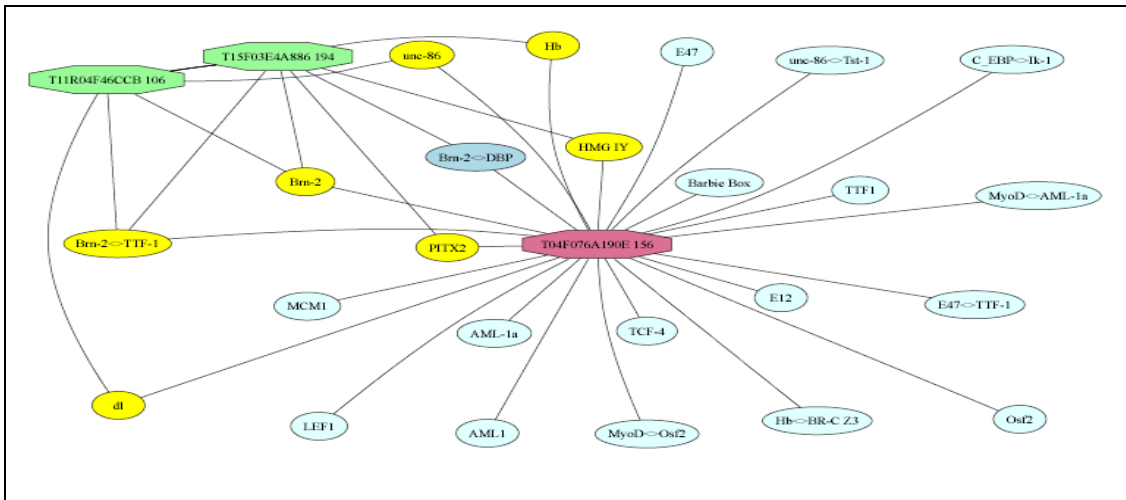


Figure 6.3 Relation network for genes and TFBSs

This network is generated following a typical question that biologist may have in transcription regulation. A specific gene (T04F076A190E) is selected together with some number (in our case seven) TFBSs of specific interest. We want to find out other genes that contain the same set of TFBSs and also to see other information of relevance to transcription regulation.

Below we provide explanation of the content of network depicted in Figure 6.3.

Octagon Nodes

The nodes presented as octagons correspond to the genes. They may appear in two colors.

- Violet color → user specified gene; for example T04F076A190E in Fig 6.3

- Green color → other genes found that share with the originally selected gene in their promoters the set of seven TFBSs required by the user.

Ellipse Nodes

Ellipse nodes correspond to TFBSs that are found in the promoters of genes present in the graph. They also may appear in different colors.

- Yellow color → TFBS is one of the TFBSs that the user has selected and it is shared by some other genes. Examples in Fig.6.3 are dl, PITX2 and Hb, etc. All these PEs control the transcription of T15F03E4A886, T11R04F46CCB, and T04F076A190E genes.
- Navy color → TFBS that is common to the presented genes but was not specified by the user in the specific list box 3.b; such as Brn2 <> DBP.
- Light blue color → TFBS found in promoter of the user selected gene, but it is not shared by the other genes found by the system. Examples are LEF1 and E12 which control the transcription of the T04F076A190E, but not the other two genes.

In the layout, all relations and characteristics between TFBSs and genes are given in terms of different features and interconnection. This visual presentation provides the more comprehensive and convenient insight into the relation of TFBSs and genes than it could be possible to get using tabular approach. Moreover, user can change the selection of genes and TFBSs and inspect different network of interconnections.

6.6 Program Flow chart

Here we present in Figure 6.4 the flowchart of the program implemented in the system and describe functions of each component block. Generally it can be specified as the instruction and data block.

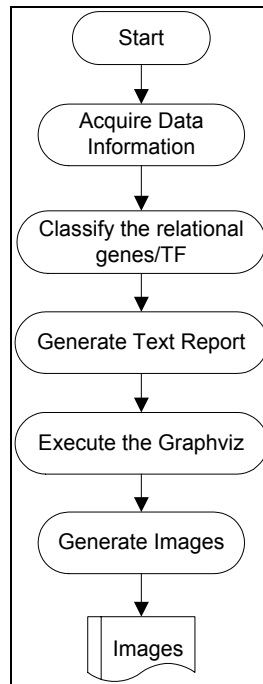


Figure 6.4 Program flow chart for TFMapper

Main Instruction block is to illustrate the program procedures to acquire data and visualize the network of gene and their associated TFBSs.

- **Acquire Data information:** This block acquires PEs/genes information from the input files, whose format is as shown in Fig 4.1.
- **Classify the relation of gene/TF:** This block collects and classifies PEs based on their association to genes, and select PEs and associated gene.
- **Generate the Text Report:** This block produces the report for PEs and their controlled genes on one text report, which follows the format of Fig 6.2.

- **Execute the Graphviz:** This block calls the Graphviz package with the image parameters setting to produce graphic layout.
- **Generate Images:** This block converts the text report into graphic images, for example Fig 6.2, to the image file, as shown in the Fig 6.3.

File information consists of the input data file imported by the user, and the output one is the images file for the network.

- **Images File:** The image, contains the connection of genes and the associated PEs, as shown in Fig 6.3.

6.7 Applications of TFMapper

This section is based on an unpublished study [76]. Here, we show how visual presentation of network data can be useful in analyzing complex relations between genes and their transcriptional regulators. We will illustrate this on the example of epithelial ovarian cancer. This cancer is one of the most deadly gynecological cancers and there is no cure for it yet. If it is not diagnosed early, the mortality is rather high [64]. Moreover, it is very difficult to diagnose it early. Thus, it is of interest to search the effects that epithelial ovarian cancer may cause and through these effects to attempt to identify genes that are involved. Since these genes are never active alone, but are always part of bigger gene networks, it is of interest to find out those genes that are likely to be co-regulated in epithelial ovarian cancer. Such genes could be good targets for further investigation as potential diagnostic markers or even as drug targets.

We used a recent microarray study of gene expression in patients with epithelial ovarian cancer [65]. From all genes expressed, we selected those that were very highly

expressed (more than 5 fold). There were in total 19 such genes. For these genes we determined promoters by using H-invitational database [74]. We were able to find promoters for 17 out of 19 genes initially selected. Then we determined promoters covering region [-800,+200] relative to the estimated TSS. Using all available matrix models for TFBSs contained in TRANSFAC Professional database ver. 7.4 [67] and mapped them to the promoter sequences. The thresholds for mapping were based on the *minSUM* profiles for matrix models [54]. These thresholds in are optimized to provide the minimum sum of false positive and false negative predictions of binding sites. Then, the promoter content of 17 overexpressed genes was compared to that of human promoters from H-invitational database. We determined the over-representation index (ORI) using method from [57]. All TFBS mapped to promoters were ranked according to decreasing ORI values. We used for annotation of 17 promoters only those TFBSs that had $\text{ORI} \geq 1.5$. The resulting file is given in Appendix 1.

Then we generated a network of all genes from the set of 17 highly expressed that have at least five common PEs. This network is given in Fig. 6.5.

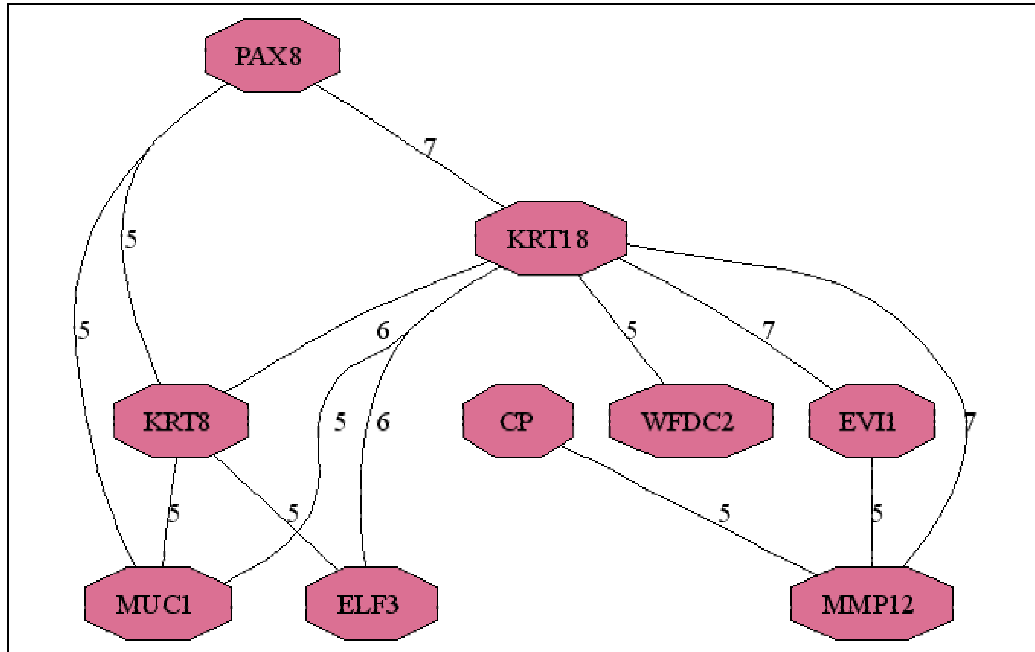


Figure 6.5. A subnetwork of interconnected genes from group of 17 very highly expressed in epithelial ovarian cancers. The link between the genes is made only if they share at least five PEs in their promoters.

Although there are no rules how to group genes into subnetworks we can observe that KRT18 and KRT8 (genes from keratin group) form one small network with attached PAX8, ELF3, MUC1 and WFDC2 genes. The other small network could be the one around MMP12 (matrix metalloproteinase 12) gene that associates CP and EVI1. In this consideration we looked also into the functionality of these genes. Keratin genes from the group of 17 genes we considered (KRT8, KRT13, KRT18) are well known for their involvement in the integrity of epithelial cells and, moreover, they are implicated in epithelial cancers [66]. Matrix metalloproteinase genes from the group of 17 genes (MMP9, MMP10, MMP12) are involved in destruction of extracellular matrix during normal physiological processes but also in diseases and cancer metastasis [75]. These two

gene groups (keratins and matrix metalloproteinase) are involved in different processes relative to cancer state and thus are likely to be part of different gene regulatory networks.

In order to find out what in more details what are members of such potential two subnetworks and to try to infer what may be their transcriptional regulators, we applied TFMapper to the 17 genes we analyzed. The gene networks are presented in Fig. 6.6 and Fig. 6.7.

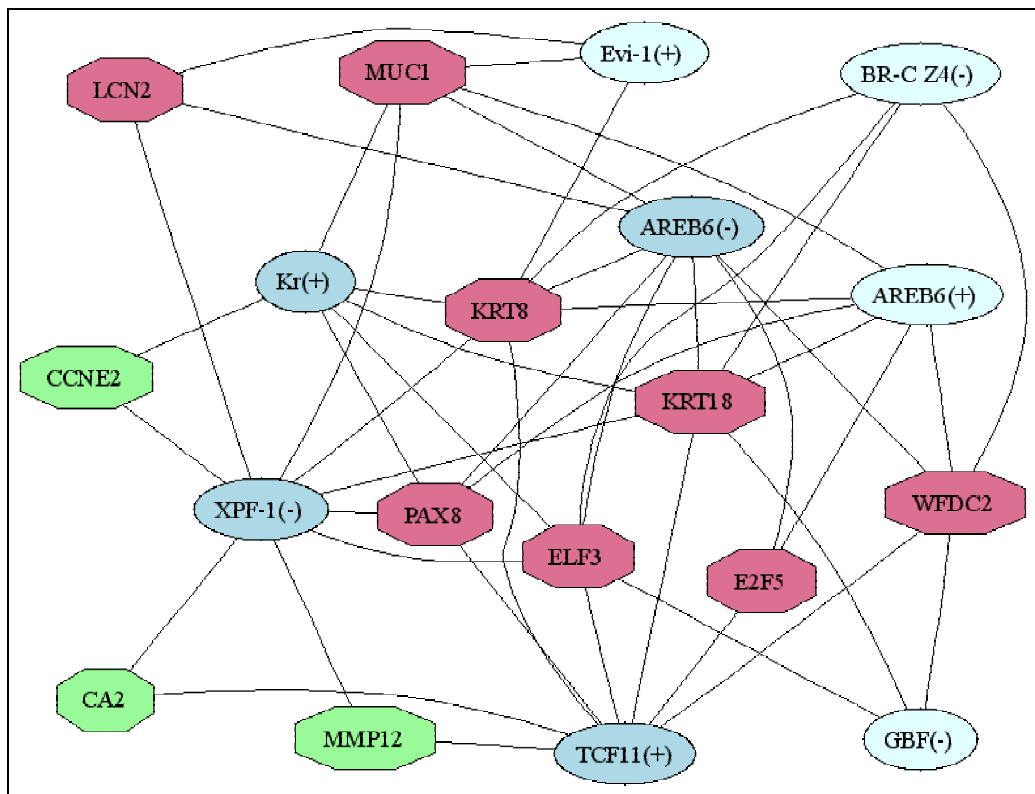


Figure 6.6. The network of genes that are highly expressed in epithelial ovarian cancer shown with PEs that potentially control these genes. The network is generated by TFMapper using four PEs (TCF11(+), AREB6(-), XPF-1(-), Kr(+)) as seed PEs.

Analysis of the keratin gene group (in our case KRT8 and KRT18) subnetwork (Fig.6.6) and their specificity that do not appear in matrix metallopeptidase group (MMP9 and MMP12) (Fig.6.7), revealed that the keratin group contains PE AREB6(-) (i.e. AREB6 binding site on ‘-‘ strand) that does not appear in matrix metallopeptidase group. For this reason it appears that keratin promoters have as a characteristic feature [96] AREB6(-) and at least one other PE such as TCF11(+), XPF-1(-) and Kr(+). Thus, the set of genes that associate with keratin group are LCN2, MUC1, WFDC2, ELF3, PAX8, E2F5. All six genes are implicated in various cancers [68, 69, 70, 71, 72, 73]. The last three genes (ELF3, PAX8, E2F5) are TFs for themselves.

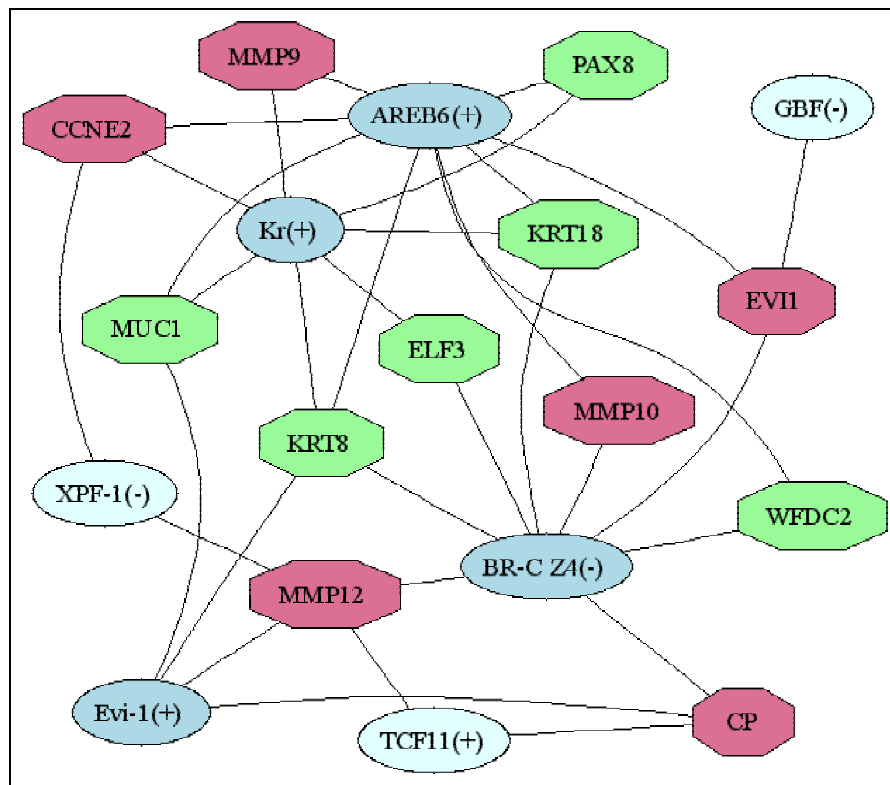


Figure 6.7. A larger gene network that contain a subnetwork of genes associated with matrix metallopeptidase group.

For matrix metalloproteinase related network (Figure X3) we can observe that the associated genes (that are not part of the keratin group) have very different promoter content characterized by either AREB6(+) and BR-C Z4(-), or Evi-1(+) and XPF-1(-), thus including CCNE2, CP and EVI1 genes.

The purpose of this section was to show how graphical presentation can help in analysis of associations of complex structures such as gene regulatory networks in a case of one particular disease. It is interesting to note that graphical presentation of potential associations of genes through their PEs revealed a lot of information that will be difficult to infer from tabular presentation.

Chapter 7 Discussions and Comments

7.1 Heuristic System Performance

The heuristic methods and the statistical parameters, which are discussed in Chapter 3, have been integrated and implemented as one motif prediction system, Dragon MotifBuilder [27]. This system was developed with C language, and it could be supported in different operating system. System performances, in term of efficiency and precision, are discussed in this chapter.

Before we compared our system with other systems, we performed a similar analysis on three motif search algorithms for human histone promoters [56] on different systems discussed on Chapter 3. The results suggest that MEME may provide more accurate predication of regulatory elements than the other two programs, AlignACE and CONSENSUS. AlignACE has a better speed performance over other two programs on simulated samples. However, all these search models are developed on the local alignment principle. Therefore, MEME is considered as one of the best performance system for the motif detection. Thus we did the comparison on motif detection by evaluating MEME and our DMB with same dataset.

7.1.1 Efficiency

The aim of motif prediction system is to provide the efficient solution to obtain the homogeneous groups of motif in large subsets of sequences in reasonable time. This homogeneous group of motifs can be helpful to predict the new motifs. To evaluate this

purpose, the system is applied to analyze a set of 8694 promoter sequences in double-stranded search, which covers [+1,+100] relative to transcription start sites, and locate the top 20 ranking motif with high average information content. The experiments were evaluated on a Window XP PC with 1.8 GHz processor and 512 Mbytes memory.

The criteria and results for the search experiment are:

Table 7.1 Search criteria for EM and GA for comparison

	Expectation Maximization	Genetic Algorithm
random initial	Yes	Yes
Threshold	expected $\theta_{th} = 0.82$	mismatch = 1
No. motif	20	20
motif length (nts)	8-12	9
motif occurrence	one/zero per sequence	one/zero per sequence
HMM	Yes	No
p-value	Yes	No
e-value	Yes	No
time (mins)	456	918
sequence coverage	8332/8694	4568/8694
IC range	8.96 – 15.85	13.44 -16.52

Generally, the EM has more efficient searching feature, and high population of the motif. However, the GA shows good characteristic of information content. All of these features are determined by the characteristic of the algorithms. The EM is one method to achieve local optimal point, but GA is a global search technique. So the EM always terminates once it could get the optimal motif. But the characteristics of GA, such as mutation and cross over, allow the new element to break through the local optimal point. Therefore, it may take GA more time to obtain one global optimal point.

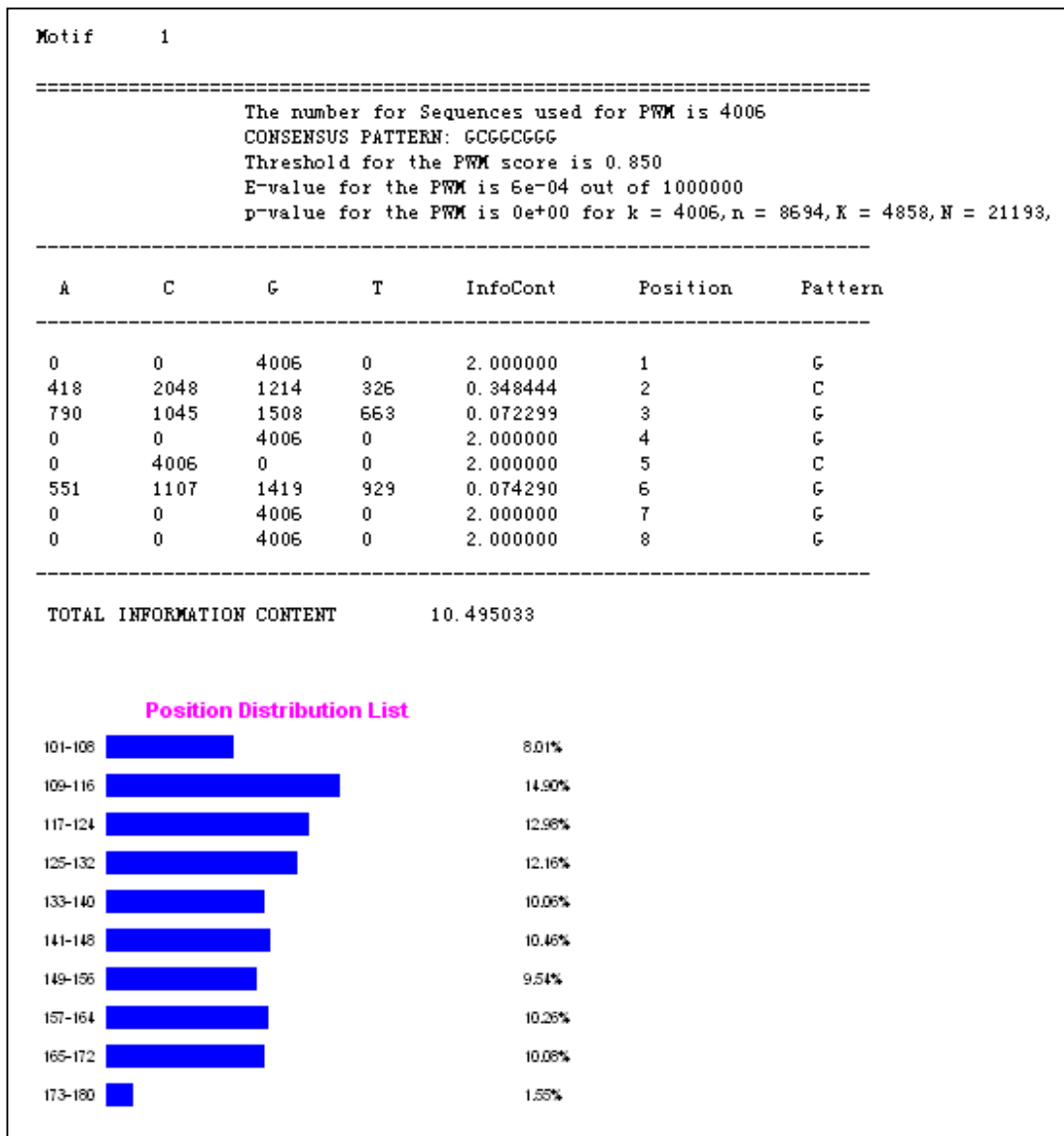


Figure 7.1 Report for motifs obtained

For illustration purposes, the snapshot of fragment of the EM search report was shown as the Fig 7.1. The summary information of this motif group contains the binding site of GC box. It is present in 4006 out of 8694 sequences, and appears significant biased on the sequence position. The high information content suggested that the motif group is highly homogeneous. Moreover, the homogenous motif group appears to have completely conserved nucleotides at 6 positions.

Some other experiments [27], also shows that the algorithms are rather efficient to analyze the large scale dataset, such as 18,326 human promoter sequences with more than 54 million nucleotides. Additionally, on average, about 25% of motifs found with the software do not belong to the already known transcription factor binding sites and represent the potentially new binding sites in the analyzed promoters.

7.1.2 Precision

Besides the speed of processing data, the precision of the prediction is also one of the most important parameters in the system. So the comparison was experimented between our software and MEME. MEME is considered as one of the currently best motif discovery systems in term of specificity and sensitivity. So MEME was selected as the candidate to compare with our tool.

In the experiments, promoter sequences from two antimicrobial peptide families: Cathelicidin and Proenkaphalin were considered and we compared the motifs found in these two families based on the motif search programs. We were able to determine precise promoters for three ortholog sequences in each antimicrobial peptide family. These sequences are selected from human, mouse and rat. Experimental studies and prior TFBS predictions for cathelicidin promoters for the human, mouse and rat species report presence of NF-kappaB, NF-IL6, LF-A1, NFI, TCF, VDR,Sp1, AP2, PU.1, IL-6-RE binding sites. For proenkaphalin, the reported functional sites were: AP-1, NF1, TATA, AP2, NF-KB, MZF-1, NF-Y.

The criteria and results are shown as following:

Table 7.2 Search criteria for MEME, EM and GA for comparison

	MEME	Expectation Maximization	Genetic Algorithm
Random initial	N.A.	Yes	Yes
threshold	N.A.	expected $\theta_{th} = 0.88$	mismatch = 1
No. motif	20	10	20
motif length (nts)	10-15	10-15	10
cathelicidin	4/9	6/9	3/9
proenkaphalin	3/9	7/9	4/9

The computation results show better prediction accuracy among the different systems with the experimental ones, although DMB with expectation Maximization algorithm has detected the largest proportion of previously known TFBSs. As a conclusion, the predicting accuracy of these systems does not change significantly for different family of promoters.

7.2 Comments on graphical representation

Three different graphical representation approaches have been developed to visualize the promoter content data, and assist biologists to capture the information effectively, and improve the quality of analysis.

All the software possesses the following features:

1) Interactivity:

The interactive presentation is an impressed and effective approach; especially in web-based graphical applications. The JavaScript is used to enhance such interactive presentation. The graphical images for viewing the promoter

structure provided pop up window according to the on mouse effect. The popup message contains all the information about Transcription Factor Binding site, such as strand location and overrepresentation.

2) Effective representation:

The main goal for motif visualization is to provide effective representation contrast to classical text format information. The protein-motif network layout simplifies the complicated correlations between the promoters, and presents further motif information, which might not be aware by the analyzers at the initial stage. The motif position distribution chart is illustrated in the heuristic algorithms, which could further detect consensus pattern with the position significance assumption. Moreover, different colors used in the images are easy for user to identify the range of overrepresentation and observe the factors systematically.

3) Flexibility for multiple operational systems

The visualization tools are requested to operate in different operational system. So the programs were developed with perl or C languages, which could be supported by different operatingl systems. The compatibility and flexibility of the programs for different systems also allows it to function in the web-based application, which is one of the main utility for bioinformatics.

Chapter 8 Conclusion and Further work

The completion of the Human Genome Project in 2003 has generated a huge volume of the genome sequences and produced vast quantity of biological information, which lead scientists to recognize the importance to present and characterize these data. Visual representation of biological data offers more convenient and more suitable insight into data to facilitate human interpretation, because it can provide human-readable diagrammatic visualization of relations from the ambiguous data. Even though great effort has been invested into graphical representation of information in bioinformatics [[13](#), [14](#), [40](#), [41](#)], the difficulties still remain in the presentation of the biological information due the complexity of the connection between entities [[41](#)].

Therefore, this study focused on exploring the suitable ways to present specific types of biological information as graphic. This specific information is related to PEs, and more broadly to transcription regulation. It is tightly associated with methods to generate data that can enable such graphical presentation. Moreover, it is also essential for us to develop systems to prepare the data for this purpose. Thus, in our work, we have developed several convenient ways for suitable presentation of specific, transcription regulation related, biological information, and developed some simple but effective presentation methods to enrich the biological content by visualizing the TFBSs/motifs, composition of promoters and their associated genes. Moreover, we have developed one accurate and efficient motif search application with the heuristic algorithms.

In the graphical presentation of biological information, we have attempted different approaches by utilizing various graphical packages to express the transcription regulatory relation.

- One graphical interface database (Dragon REGHSdb), which describes the transcriptional regulatory motifs in the promoter region, has been developed with the graphical tool TSSViewer. This database with such visual representation of promoter content enables different means for biologist to get insight into promoter structure of his target gene groups, which cannot be provided in the traditional tabular expression.
- Another system, DMB, has been developed to generate the graphical report for PEs, which are obtained in the heuristic algorithm. This system has been published as the web-application, which allows the users to easily figure out the putative TFBSs, their cumulative distribution and distribution along individual sequences with the graphical approach..
- TFMapper was developed as one effective solution to generate small-size transcriptional regulatory networks suitable for the analysis of roles of PEs in control of various genes. This visual presentation provides the more comprehensive and convenient insight into the relation of TFBSs and genes than it could be possible to get using tabular approach.

For preparing the data for the graphic presentation, we have developed the efficient heuristic methods to detect the homogenous motif groups in large scale biological sequence sets, and applied the statistic measures for selection of the motifs.

This system has been compared with other systems, such as MEME [9, 10]. It achieves better performance in term of accuracy and speed than the other methods. Therefore, the shortlist motif groups, which are obtained by this system, would be helpful for the biologists to identify and discover the transcription information easily.

Even though the graphic representation provides a comprehensive and simply presentation for the TFBSs and their associated gene, it is lack of the interactive animation which could enhance the presentation effect. Thus, some other interactive graphical packages, such as SBML [59] and VRML [49], may be considered as the further development tool to complement this. Moreover, the heuristic algorithms in the data preparation system are still sensitive to certain parameters setting, which affect the accuracy of the predication. For example, EM is quite sensitive to the initial maxima obtained in the search, and it sometimes stops searching when it reaches a local maxima. In order to overcome the sensitivity and maintain the stability of the search, more complicated statistical and possibility model should be implemented as part of the heuristics algorithm.

However, the preparation and presentation of the biological information is not a simple computer science topic. It required a deep understand on the biological problems. Moreover, no universal solution has been established for all the problems. But in our study, we have successfully developed the graphical presentation systems, which cater for presentation of various TFBSs/motifs, promoters and their associated genes. Additionally, the data preparation system, DMB, was developed and evaluated as one of the precise and efficient system to identify the homogenous motifs. All the work we have

done is only a start, we need continue exploring the approaches to present and characterize the biological data.

References

1. Attwood T. K, Parry-Smith D. J (1999) Introduction to bioinformatics, Prentice Hall
2. Fickett .J.W, Hatzigeorgiou A.G (1997) Eukaryotic promoter recognition Genome Research.:7:861-878
3. Alberts, Bruce, A. Johnson, J. Lewis, Raff .M, Roberts K, and Walter P (2002), Molecular Biology of the Cell. Fourth Edition. New York: Garland.
4. http://www.blc.arizona.edu/Molecular_Graphics/DNA_Structure/DNA_Tutorial.HTML
5. <http://en.wikipedia.org/wiki/Chromosomes>
6. <http://www.web-books.com/MoBio/Free/Ch3F.htm>
7. Werner T (1999) Models for prediction and recognition of eukaryotic promoters, Mamalian Genome, 10:165-168
8. Boysen C, Simon M.I, Hood L.E (1997). Analysis of the 1.1 M-b human alpha/delta T-cell receptor locus with bacterial artificia chromosome clones. Genome Research 7:330-338
9. Bailey T.L, Elkan C (1994). Fitting a mixture model by epxectation maximization to discover motifs in biolpolymers. In Proceedings of the 2nd International Conference on Intelligent Systems for Molecular Biology, Vol. 2: 28-36. AAAI Press.
10. Bailey T.L, Elkan C (1995). Unsupervised learning of multiple motifs in biopolymers using expectation maximization. Machine Learning, 21:51-80.
11. Hughes J .D, Estep .P .W, Tavazoie S, Church G.M (2000). Computational identification of Cis-regulatory elemtens associated with groups of functionally

- related genes in *Saccharomyces cerevisiae*. *Journal. Molecular. Biology* Vol. 296:1205-1214
12. Helden J. V, Andre B, Collado-Vides J, (2000) A web site for the computational analysis of yeast regulatory sequences. *Yeast*, Vol.16: 177–187.
 13. Sandelin A, Alkema W, Engstrom P, Wasserman W.W, Lenhard B (2004), JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Research* Vol 32: 91-94
 14. Pan H, Zuo L, Choudhary V, Zhang Z, Leow SH, Chong FT, Huang Y, Ong VW, Mohanty B, Tan SL, Krishnan SP, Bajic VB. (2004) Dragon TF Association Miner: a system for exploring transcription factor associations through text-mining. *Nucleic Acids Res.* 2004 Jul 1;32 (Web Server Issue): 230-234
 15. Parra G, Agarwal P, Abril J.F, Wiehe. T, Fickett J.W, Guigo. Roderic. (2003) Comparative gene prediction in human and mouse. *Genome Research*. Jan; 13, 108-117
 16. Bilmes. J (1997) A gentle tutorial on the EM algorithm and its application to parameter estimation for gaussian mixture and hidden markov models, Technical Report, University of Berkeley, ICSI-TR-97-021
 17. Smola. A, Moon TK (1996) The expectation-maximization algorithm, *IEEE Trans Signal Processing*. 1996 Nov; 47-60
 18. Lafferty. J. Notes on the EM algorithm. Online article. (<http://www.cs.cmu.edu/afs/cs.cmu.edu/academic/class/11761-s97/www.tex/em.ps>)
 19. McLachlan .G.J, Krishnan T. (1997) *The EM Algorithm and Extensions*. John Wiley and Sons, Inc.

20. L Yang, E Huang, VB Bajic (2004), Some implementation issues of heuristic methods for motif extraction from DNA sequences, International.Journal.of Computing System.Signals, 5(2)
21. R. Dugad and U.B. Desai (1996), "A Tutorial on Hidden Markov Models," Published Online. <http://vision.ai.uiuc.edu/dugad/guestbook/addHMMguest.html>. May 1996.
22. <http://www.ch.embnet.org/CoursEMBnet/Exercises/statistics.html>
23. <http://www.people.virginia.edu/~wrp/cshl98/Altschul/Altschul-1.html#ref10>
24. A. Krogh, M. Brown, I. S. Mian, K. Sjolander, and D. Haussler (1994), Hidden Markov models in computational biology: Applications to protein modeling. Journal.of Molecular. Biology , 235:1501--1531, February
25. <http://www.isixsigma.com/dictionary/P-Value-301.htm>
26. Beyer, W. H. CRC (1987) Standard Mathematical Tables, 28th ed. Boca Raton, FL: CRC Press, pp. 532-533.
27. E Huang, L Yang, R Chowdhary, A Kassim, VB Bajic (2005), An algorithm for ab initio DNA motif detection, Chapter 4 in Information Processing and Living Systems, World Scientific, 611-614,
28. F Glover (1989) Tabu Search - Part I. ORSA Journal on Computing, 1: 190-206
29. F Glover (1990) Tabu Search - Part II. ORSA Journal on Computing, 2: 4-32
30. F Glover, M Laguna (1997) Tabu Search, Kluwer Academic Publisher
31. RW Eglese (1990) Simulated annealing: a tool for operational research, European Journal of operational Research, Vol. 46, No. 3. June 15: 271 – 281.
32. M Fleischer (1995) Simulated annealing: past, present, and future, pages: 155 – 161, ACM Press, New York, NY, USA

33. S Kirkpatrick, C Gelatt, M. Vecchi (1983) Optimization by Simulated Annealing. *Science*, 220(4598): 671-680.
34. D.S Johnson, C.R Aragon, LA McGeoch, C Schevon. (1989) Optimization by Simulated Annealing: An Experimental Evaluation. *Operations Research*, 37(6): 865-892.
35. C Tovey. (1988) Simulated annealing. *American Journal of Mathematical and Management Sciences*, 8(3&4): 389-407.
36. D.W Mount. (2001). *Bioinformatics: Sequence and Genome Analysis* , Cold Spring Harbor Laboratory Press, New York. Chapter 2 & 3.
37. E.R. Gansner (2004). Drawing graphs with Graphviz. <http://www.graphviz.org/Documentation.php>
38. E. Segal, M. Shapira, A. Regeve, D. Pe'er, D. Bostein, D. Koller and N. Friedman (2003), Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, volume 34, p. 166-176
39. Kohn KW, et.al Molecular Interaction Maps of Bioregulatory Networks: A General Rubric for Systems Biology, *Mol Biol Cell*. 2005 Nov 2
40. Kitano, H.(2003), A Graphical Notation for Biological Networks. *BioSilico*, 1: p.169-176.
41. Kitano, H.et.al. (2005) Using process diagrams for the graphical representation of biological networks, *Nature Biotechnology* 23(8), 961 - 966
42. D. Karolchik, R. Baertsch, M. Diekhans, T. S. Furey, A. Hinrichs, Y. T. Lu, K. M. Roskin, M. Schwartz, C. W. Sugnet, D. J. Thomas, R. J. Weber, D. Haussler and W. J.

- Kent, (2003), The UCSC Genome Browser Database, Nucleic Acids, Vol. 31 (1): 51
– 54
43. Altschul,SF., Gish,W., Miller,W., Myers,E.W and Lipman,D.J. (1990) Basic local alignment search tool. J. Mol. Biol., 215,403 -410
44. T. Hubbard, D. et al (2005), Ensembl 2005, Nucleic Acids Res. Jan Vol 33 Database issue: 447 – 453
45. Gary D. S (2000), DNA binding sites: representation and discovery. Bioinformatics.Vol. 16(1): 16 -23
46. M. Tompa et al (2005), Assessing computational tools for the discovery of transcription factor binding sites, Nature Biotechnology. Vol 23(1): 137 -144
47. G.D. Battista, P. Eades, R.Tamassia, I. G. Tollis (1999), Graph Drawing: Algorithms for the Visualization of Graphs. Prentice Hall.
48. <http://www.boutell.com/gd/>
49. Hartman, J. et al. (1996). The VRML 2.0 Handbook, Building Moving Worlds on the Web Addison Wesley.
50. R. Lea, K. Matsuda and K Miyashita (1996), Java for 3D and VRML Worlds, New Riders Publishing, Indianapolis Indiana.
51. Wong, L, et al (2001), PIES: Protein Interaction Extraction System, Pac Symp Biocomput:520-31.
52. Crooks GE, Hon G, Chandonia JM, Brenner SE ,(2004).WebLogo: A sequence logo generator, Genome Research, 14:1188-1190
53. Cavin Périer, R., Junier, T., Bucher, P.(1998). The Eukaryotic Promoter Database EPD, Nucleic Acids Res.26, 353-357.

54. Kel AE, Gossling E, Reuter I, Cheremushkin E, Kel-Margoulis OV, Wingender E. (2003). MATCH: A tool for searching transcription factor binding sites in DNA sequences, *Nucleic Acids Research*. July 1;31(13):3576-3579.
55. <http://www.gene-regulation.com/pub/databases.html>
56. R Chowdhary, R. Ayesha Ali, W Albig, D Doenecke and VB Bajic (2005), Promoter modeling: The case study of mammalian histone promoters, *Bioinformatics*, 21(11):2623-2628
57. VB Bajic, V Choudhary, CK Hock, Content analysis of the core promoter region of human genes, *In Silico Biology*, 4:109-125, 2004
58. Son MY, Kim TJ, Kweon KI, Park JI, Park C, Lee YC, No Z, Ahn JW, Yoon WH, Park SK, Lim K, Hwang BD (2002), ATF is important to late S phase-dependent regulation of DNA topoisomerase IIalpha gene expression in HeLa cells, *Cancer Letter* Vol. 184(1):81-88
59. Hucka M, Finney A, et al. (2003), The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models, *Bioinformatics*. March 1; Vol. 19(4):524-31
60. <http://www.opengl.org/>
61. Klaus-Peter Fahlbusch, Thomas D. Roser (1995), HP PE/SolidDesigner: dynamic modeling for three-dimensional computer-aided design; *Hewlett-Packard Journal*
62. <http://usa.autodesk.com/adsk/servlet/index?siteID=123112&id=2704278>
63. S B. Montgomery, et al, (2004) Sockeye: A 3D Environment for Comparative Genomics, *Genome Research* Vol.14:956-962

64. Cannistra SA, (2004) Cancer of the ovary, *New England Journal of Medicine*, 351, 2519-2529
65. Shridhar,V. et al. Genetic analysis of early- versus late-stage ovarian tumors, *Cancer Research* 61, 5895 – 5904
66. Trask, D.K., Band, V., Zajchowski, D.A., Yaswen, P., Suh, T., and Sager, R. (1990) Keratins as markers that distinguish normal and tumor-derived mammary epithelial cells. *Proceedings of the National Academy of Sciences, USA* 87: 2319-2323
67. Matys, V et al (2003). TRANSFAC (R): transcriptional regulation, from patterns to profiles. *Nucleic Acids Research*. Vol. 31, 374-378.
68. Croce,M.V et.al (2003) Tissue and serum MUC1 mucin detection in breast cancer patients. *Breast Cancer Research Treat.* 2003 Oct;81(3):195-207
69. Hellstrom I, Raycraft J, Hayden-Ledbetter M, et al (2003). The HE4 (WFDC2) protein is a biomarker for ovarian carcinoma. *Cancer Research*, 63: 3695-3700
70. Hanai, J. et al. (2005) Lipocalin 2 Diminishes Invasiveness and Metastasis of Ras-transformed Cells, *Journal of Biological Chemistry*, 280 13641-13647
71. Vaishnav, Y.N. et al (1999) Differential regulation of E2F transcription factors by p53 tumor suppressor protein, *DNA Cell Biology* 18, 911-922
72. Kroll TG, Sarraf P, Pecciarini L, et al.(2000) PAX8-PPAR γ 1 fusion oncogene in human thyroid carcinoma. *Science*; 289:1357-60
73. Galang, C.K., MullerW.J., Foos,G.,Oshima, R.G, Hauser, C.A. (2004) Changes in the expression of many Ets family transcription factors and of potential target genes in normal mammary tissue and tumors, *Journal of Biolgocial Chemistry* 279, 11281-11292

74. Imanishi T. et al. (2004) Integrative annotation of 21,037 human genes validated by full-length cDNA clones. *PLoS Biol.* Jun;2(6):e162. Epub 2004 Apr 20.
75. Hijova E. Matrix metalloproteinases: their biological functions and clinical implications. *Bratisl Lek Listy.* 2005;106(3):127-32.
76. K Narasimhan, VB Bajic, Ma Choolani (2005) Unpublished result. E2F5 in blood: potential marker for epithelial ovarian cancer.

Appendix 1:

```
StartPos:-800 EndPos: 200
MinCoef:1.537900 MaxCoef:2.968200
=====
Hsu2000d500_1381_res.mh
>mRNA|PAX8|
X69699;S77906;S77905;S77904;NM_013992;NM_013953;NM_013952;NM_013951;NM_
003466;L19606;BC001060|LocusID|7849|Chromosome|2|Strand|-
|Tss|10001(114131642)|ChroPos|114130643-114141643|length|11000
=====
-1 XPF-1" -716..-707 1.968300 10
+1 Kr" -612..-603 2.292000 7
-1 XPF-1" -591..-582 1.968300 10
+1 TCF11" -501..-489 1.982600 10
-1 AREB6" -186..-178 1.561700 8
+1 AREB6" -164..-156 2.968200 10
+1 Kr" -36..-27 2.292000 7

=====
Hsu2000d500_1676_res.mh
>mRNA|CA2|Y00339;NM_000067;M36532;J03037;BC011949|LocusID|760|Chromosom
e|8|Strand|+|Tss|10001(86450886)|ChroPos|86440886-86451886|length|11000
=====
+1 Evi-1" -796..-782 1.537900 7
-1 XPF-1" -731..-722 1.968300 10
+1 TCF11" -583..-571 1.982600 10
-1 XPF-1" -349..-340 1.968300 10

=====
Hsu2000d500_195_res.mh
>mRNA|KRT18|X12883;X12881;X12876;NM_199187;NM_000224;CD106591;BG753529;
BC020982;BC009754;BC008636;BC004253;BC000698;BC000180;AK129587|LocusID|
3875|Chromosome|12|Strand|+|Tss|10001(51628906)|ChroPos|51618906-
51629906|length|11000
=====
-1 XPF-1" -793..-784 1.968300 10
+1 Kr" -726..-717 2.292000 7
-1 AREB6" -593..-582 1.561700 8
-1 GBF" -383..-375 2.073100 4
-1 BR-C Z4" -367..-355 2.220100 8
-1 BR-C Z4" -362..-350 2.220100 8
-1 BR-C Z4" -361..-349 2.220100 8
-1 BR-C Z4" -360..-348 2.220100 8
-1 BR-C Z4" -359..-347 2.220100 8
-1 BR-C Z4" -358..-346 2.220100 8
-1 BR-C Z4" -357..-345 2.220100 8
-1 BR-C Z4" -356..-344 2.220100 8
-1 BR-C Z4" -355..-343 2.220100 8
-1 BR-C Z4" -354..-342 2.220100 8
+1 Kr" -178..-169 2.292000 7
-1 XPF-1" -145..-136 1.968300 10
+1 TCF11" -5..8 1.982600 10
+1 AREB6" 70..78 2.968200 10
```

=====
Hsu2000d500_1963_res.mh
>mRNA|WFDC2|X63187;NM_080736;NM_080735;NM_080734;NM_080733;NM_006103;AF330262;AF330261;AF330260;AF330259|LocusID|10406|Chromosome|20|Strand|+|Tss|10001(44783802)|ChroPos|44773802-44784802|length|11000
=====

+1	TCF11"	-688..-676	1.982600	10
-1	GBF"	-556..-548	2.073100	4
+1	AREB6"	-510..-502	2.968200	10
-1	BR-C Z4"	-492..-480	2.220100	8
-1	GBF"	-390..-382	2.073100	4
+1	AREB6"	-2..10	2.968200	10
-1	AREB6"	121..132	1.561700	8

=====
Hsu2000d500_245_res.mh
>mRNA|LCN2|X83006;NM_005564;CA454137;BX644845;BF354583;BC033089;AW778875|LocusID|3934|Chromosome|9|Strand|+|Tss|10001(126287762)|ChroPos|12627762-126288762|length|11000
=====

+1	Evi-1"	-789..-775	1.537900	7
-1	AREB6"	-751..-740	1.561700	8
-1	XPF-1"	-746..-737	1.968300	10
-1	AREB6"	-398..-390	1.561700	8
-1	AREB6"	-224..-216	1.561700	8
-1	AREB6"	-118..-107	1.561700	8
-1	AREB6"	-4..9	1.561700	8

=====
Hsu2000d500_257_res.mh
>mRNA|KRT8|X98614;X74929;X12882;U76549;NM_002273;M77025;M34225;M26512;BC063513;BC011373;BC008200;BC000654|LocusID|3856|Chromosome|12|Strand|-|Tss|10001(51585106)|ChroPos|51584107-51595107|length|11000
=====

-1	BR-C Z4"	-644..-632	2.220100	8
+1	TCF11"	-501..-489	1.982600	10
+1	Evi-1"	-273..-259	1.537900	7
+1	AREB6"	-149..-137	2.968200	10
+1	Kr"	-116..-107	2.292000	7
-1	AREB6"	-82..-70	1.561700	8
-1	XPF-1"	185..194	1.968300	10

=====
Hsu2000d500_2599_res.mh
>mRNA|CP|X04136;NM_000096;M13699;M13536;AK095290|LocusID|1356|Chromosome|3|Strand|-|Tss|10001(150260501)|ChroPos|150259502-150270502|length|11000
=====

+1	TCF11"	-760..-748	1.982600	10
+1	Evi-1"	-696..-682	1.537900	7
-1	BR-C Z4"	-684..-672	2.220100	8
+1	Evi-1"	-656..-642	1.537900	7
+1	Evi-1"	-484..-470	1.537900	7
+1	TCF11"	-437..-425	1.982600	10
-1	BR-C Z4"	-126..-114	2.220100	8

```

Hsu2000d500_2874_res.mh
>mRNA|E2F5|Z78409;X86097;U31556;NM_001951|LocusID|1875|Chromosome|8|Str
and|+|Tss|10001(86164279)|ChroPos|86154279-86165279|length|11000
=====
+1   TCF11"                -635..-623          1.982600            10
-1   AREB6"                -584..-576          1.561700            8
+1   AREB6"                -118..-107          2.968200            10

=====
Hsu2000d500_2947_res.mh
>mRNA|ELF3|U97156;U73844;U73843;U66894;NM_004433;BX537368;BC003569;AF51
7841;AF017307;AF016295|LocusID|1999|Chromosome|1|Strand|+|Tss|10001(199
265329)|ChroPos|199255329-199266329|length|11000
=====
-1   BR-C Z4"              -671..-659          2.220100            8
-1   AREB6"                -645..-637          1.561700            8
+1   TCF11"                -442..-430          1.982600            10
-1   XPF-1"                -109..-100          1.968300            10
+1   Kr"                   41..50              2.292000            7
-1   GBF"                  131..139            2.073100            4

=====
Hsu2000d500_2972_res.mh
>mRNA|EVI1|X54989;S82592;NM_005241;BX647613;BX640908;BC031019;AK025934;
AF487424;AF487423;AF164157;AF164155;AF164154|LocusID|2122|Chromosome|3|
Strand|-|Tss|10001(170185005)|ChroPos|170184006-170195006|length|11000
=====
-1   GBF"                  -746..-738          2.073100            4
-1   BR-C Z4"              -568..-556          2.220100            8
-1   BR-C Z4"              -484..-472          2.220100            8
+1   AREB6"                -335..-327          2.968200            10
+1   AREB6"                -91..-83            2.968200            10
-1   BR-C Z4"              87..99              2.220100            8
-1   BR-C Z4"              92..104             2.220100            8
-1   BR-C Z4"              154..166            2.220100            8

=====
Hsu2000d500_339_res.mh
>mRNA|MMP10|X07820;NM_002425;BT007442;BC002591|LocusID|4319|Chromosome|
11|Strand|-|Tss|10001(102189075)|ChroPos|102188076-
102199076|length|11000
=====
-1   BR-C Z4"              -441..-429          2.220100            8
+1   AREB6"                -358..-346          2.968200            10
-1   BR-C Z4"              -347..-335          2.220100            8

=====
Hsu2000d500_3600_res.mh
>mRNA|CLDN4
|NM_001305;BC000671;AK126462;AK126315;AK124076;AB000712|LocusID|1364|Ch
romosome|7|Strand|+|Tss|10001(72657289)|ChroPos|72647289-
72658289|length|11000
=====
+1   Evi-1"                -334..-320          1.537900            7
-1   XPF-1"                -49..-40            1.968300            10

=====

```

Hsu2000d500_371_res.mh

>mRNA|MUC1|X80761;X52229;X52228;U60261;U60260;U60259;NM_182741;NM_002456;M32739;M32738;J05581;AY466157;AY327600;AY327599;AY327598;AY327597;AY327596;AY327595;AY327592;AY327591;AY327590;AY327589;AY327588;AY327587;AY327586;AY327585;AY327584;AY327583;AY327582;AF348143|LocusID|4582|Chromosome|1|Strand|-|Tss|10001(152379450)|ChroPos|152378451-152389451|length|11000

-1	XPF-1"	-717..-708	1.968300	10
-1	XPF-1"	-337..-328	1.968300	10
+1	Kr"	-214..-205	2.292000	7
+1	AREB6"	-122..-111	2.968200	10
+1	AREB6"	-115..-104	2.968200	10
-1	AREB6"	88..99	1.561700	8
+1	Evi-1"	186..200	1.537900	7

Hsu2000d500_7408_res.mh

>mRNA|MMP12|NM_002426;L23808|LocusID|4321|Chromosome|11|Strand|-|Tss|10001(102283395)|ChroPos|102282396-102293396|length|11000

+1	Evi-1"	-602..-588	1.537900	7
-1	BR-C Z4"	-474..-462	2.220100	8
-1	BR-C Z4"	-436..-424	2.220100	8
-1	BR-C Z4"	-416..-404	2.220100	8
-1	BR-C Z4"	-402..-390	2.220100	8
+1	Evi-1"	-343..-329	1.537900	7
+1	TCF11"	-165..-153	1.982600	10
-1	XPF-1"	60..69	1.968300	10
-1	BR-C Z4"	152..164	2.220100	8

Hsu2000d500_7578_res.mh

>mRNA|MMP9|NM_004994;J05070;BC006093|LocusID|4318|Chromosome|20|Strand|+|Tss|10001(45322968)|ChroPos|45312968-45323968|length|11000

+1	Kr"	-621..-612	2.292000	7
+1	AREB6"	-238..-230	2.968200	10
+1	AREB6"	-1..11	2.968200	10

Hsu2000d500_7850_res.mh

>mRNA|PCNA|NM_182649;NM_002592;M15796;BU626265;BG612192;BC062439;BC000491|LocusID|51111|Chromosome|20|Strand|-|Tss|10001(5102269)|ChroPos|5101270-5112270|length|11000

+1	TCF11"	-474..-462	1.982600	10
----	--------	------------	----------	----

Hsu2000d500_9191_res.mh

>mRNA|CCNE2|NM_057749;NM_057735;NM_004702;BC020729;BC007015;AF112857;AF106690;AF102778;AF091433|LocusID|9134|Chromosome|8|Strand|-|Tss|10001(95864064)|ChroPos|95863065-95874065|length|11000

+1	AREB6"	-681..-669	2.968200	10
-1	XPF-1"	-416..-407	1.968300	10
-1	XPF-1"	-349..-340	1.968300	10

+1	Kr"	29..38	2.292000	7
-1	XPF-1"	126..135	1.968300	10

=====

SUMMARY

=====

Files Processed :17
Files having selected TFs :17
Files discarded due to N :0
Files discarded due to GC :0
GC max :1.000000 GC Min:0.000000