Analysis of Genomic and Proteomic Sequences using DSP Techniques

Raja Sekhar Kakumani

A Thesis
in
The Department
of
Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Doctor of Philosophy at
Concordia University
Montréal, Québec, Canada

March 2013

CONCORDIA UNIVERSITY
SCHOOL OF GRADUATE STUDIES

This is to certify that the thesis prepared

By: Raja Sekhar Kakumani

Entitled: Analysis of Genomic and Proteomic Sequences using DSP Techniques

and submitted in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY (Electrical & Computer Engineering)

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

Dr. G. Gouw	Chair
Dr. P. Agathoklis	External Examiner
Dr. R. Ganesan	External to Program
Dr. M.N.S. Swamy	Examiner
Dr. W. Zhu	Examiner
Dr. M.O. Ahmad	Thesis Co-Supervisor
Dr. V. Devabhaktuni	Thesis Co-Supervisor

Approved by _____
Dr. J.X. Zhang, Graduate Program Director

April 12, 2013

_____
Dr. Robin A.L. Drew, Dean
Faculty of Engineering & Computer Science

ABSTRACT

Analysis of Genomic and Proteomic Sequences using DSP Techniques

Raja Sekhar Kakumani, Ph.D.
Concordia University, 2013

Analysis of biological sequences by detecting the hidden periodicities and symbolic patterns has been an active area of research since couple of decades. The hidden periodic components and the patterns help locating the biologically relevant motifs such as protein coding regions (exons), CpG islands (CGI) and hot-spots that characterize various biological functions. The discrete nature of biological sequences has prompted many researchers to use digital signal processing (DSP) techniques for their analysis. After mapping the biological sequences to numerical sequences, various DSP techniques using digital filters, wavelets, neural networks, filter banks etc. have been developed to detect the hidden periodicities and recurring patterns in these sequences. This thesis attempts to develop effective DSP based techniques to solve some of the important problems in biological sequence analysis. Specifically, DSP techniques such as statistically optimal null filters (SONF), matched filters and neural networks based algorithms are developed for the analysis of deoxyribonucleic acid (DNA), ribonucleic acid (RNA) and protein sequences.

In the first part of this study, DNA sequences are investigated in order to identify the locations of CGIs and protein coding regions, i.e., exons. SONFs, which are known for their ability to efficiently estimate short-duration signals embedded in noise by combining the maximum signal-to-noise ratio and the least squares optimization criteria, are utilized to solve these problems. Basis sequences characterizing CGIs and exons are formulated to be used in SONF technique for solving the problems.

In the second part of this study, RNA sequences are analyzed to predict their

secondary structures. For this purpose, matched filters based on 2-dimensional convolution are developed to identify the locations of stem and loop patterns in the RNA secondary structure. The knowledge of the stem and loop patterns thus obtained are then used to predict the presence of pseudoknot, leading to the determination of the entire RNA secondary structure.

Finally, in the third part of this thesis, protein sequences are analyzed to solve the problems of predicting protein secondary structure and identifying the locations of hot-spots. For predicting the protein secondary structure a two-stage neural network scheme is developed, whereas for predicting the locations of hot-spots an SONF based approach is proposed. Hot-spots in proteins exhibit a characteristic frequency corresponding to their biological function. A basis function is formulated based on this characteristic frequency to be used in SONFs to detect the locations of hot-spots belonging to the corresponding functional group.

Extensive experiments are performed throughout the thesis to demonstrate the effectiveness and validity of the various schemes and techniques developed in this investigation. The performance of the proposed techniques is compared with that of the previously reported techniques for the analysis of biological sequences. For this purpose, the results obtained are validated using databases containing with known annotations. It is shown that the proposed schemes result in performance superior to those of some of the existing techniques.

Dedicated to my parents.

ACKNOWLEDGEMENTS

I would like to gratefully acknowledge the guidance and support of my supervisors Prof. Omair Ahmad and Prof. Vijay Kumar Devabhaktuni in the development of this research. I thank the entire staff of the Department of Electrical and Computer Engineering, especially Kimberly R. Adams and Pamela J. Fox, who have been very helpful in many ways.

My great appreciation goes to my friends, in particular Rajeev Yadav, Kaustubha Mendhurwar and Saurin Patel, for always being there for me. I would like to express my profound gratitude to my parents, my sisters Aruna and Chinna, and my parents-in-law for their unending prayers and emotional support they have provided all along. I would also like to thank my sibling Chitti, Annie and Santy for providing encouragement and confidence. I owe a huge debt of gratitude to my amazing wife, Priya, for her unconditional love, constant encouragement and enormous patience during my entire doctoral studies. More importantly, I would like to thank God for His grace, provision and support, without which my accomplishments would not have been possible.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

Acc	Performance accuracy
B	Base-pair matrix
B_u	Upper triangular matrix
D	Mask of size M
E	Model error
$G(X_n)$	SNR gain
h	Momentum
I_n	IMF output
L	Length of window
N	Length of DNA sequence X
$r(m)$	Element of R_n
R_n	Residual signal
$s(m)$	Element of S_n
Sn	Sensitivity
Sp	Specificity
$S(n)$	Log-likelihood ratio
S_n	Message signal
$\mathcal{S}(k)$	Combined magnitude spectrum
X_A	Binary indicator sequence of nucleotide A
X	DNA sequence
w_A	EIIP value of nucleotide A
X_n	Windowed sequence
$x_n(m)$	Element of X_n
X_{EIIP}	Numerical sequence using EIIP values
X_{CG}	Binary indicator sequence for C and G

$\mathcal{X}_{\mathcal{A}}(k)$	DFT of X_A
$y(m)$	Element of Y_n
Y_n	SONF output
Z_n	Output error
Δ	Difference of relative occurrence of a particular gap in CGI and non-CGI
η	Threshold
γ	Learning rate
$\iota(m)$	Element of I_n
$\phi(m)$	Element of Φ
Φ	Basis sequence
Λ_n	Scaling sequence

LIST OF ACRONYMS

3D	Three dimensional
A	Adenine
AUC	Area under the curve
C	Cytosine
CC	Correlation coefficient
CGI	CpG island
DFT	Discrete Fourier transform
DNA	Deoxyribonucleic acid
DSP	Digital signal processing
EBI	European Bioinformatics Institute
EIIP	Electron-ion interaction potential
FIR	Finite input response
FN	False negative
FP	False positive
FPR	False positive rate
G	Guanine
GUI	Graphical user interface
IIR	Infinite impulse response
IMF	Instantaneous matched filter
LS	Least squares
mRNA	Messenger RNA
MLP	Multi-layer perceptron
MWM	Maximum weighted matching
NCBI	National Center for Biotechnology Information

PDB	Protein data bank
RNA	Ribonucleic acid
ROC	Receiver operating characteristic
RRM	Resonant recognition model
SNR	Signal to noise ratio
SONF	Statistically optimal null filters
STDFT	Short-time discrete Fourier transform
T	Thymine
TN	True negative
TP	True positive
TPR	True positive rate
U	Uracil

Chapter 1

Introduction

1.1 General

Until the early nineteenth century, it was strongly believed that living matter and the inanimate matter are completely different, and hence the normal laws of chemistry were not subjected to the former. Consequently, organisms were thought to be made of chemical components unique to living creatures. In 1828, Friedrich Wohler demonstrated the conversion of ammonium cyanate, a laboratory chemical, to urea, a molecule generated by living animals. This demonstration had changed the perspective that there was something magical about the chemistry of living matters. Later, the biological macromolecule, deoxyribonucleic acid (DNA), which is now well established as a genetic material, was first discovered by Frederich Miescher in 1869, but it was nearly after a century that its true significance was revealed.

The question of how DNA could act as the genetic information was answered by James Watson and Francis Crick in 1953. They have suggested the now famous double helix structure of DNA [1], which provided a chemical basis for the genetic code, and the mechanism for DNA replication, which constitutes a basis for biological

inheritance. In 1950 Maurice Wilkins and his assistant Raymond Gosling took the first images of DNA using x-ray diffraction, which were later used by them as the basis for their structural model [2]. Unraveling the chemical basis for inheritance won Watson, Crick and Wilkins the Nobel Prize in Physiology or Medicine for 1962. This central finding drives our understanding as to how all the living cells and consequently the living organisms function.

The first complete genome of a living organism, bacterium (*Haemophilus influenzae*), was sequenced in 1995 [3]. Since then the genomes of several organisms have been completely sequenced beginning a new era of biological data acquisition and information accessibility. There are billions of nucleotides of DNA sequence data collected from thousands of organisms available in databanks such as GenBank [**?**, 4] at the National Center for Biotechnology Information (NCBI) [5], DNA Database of Japan (DDBJ) [6] and European Bioinformatics Institute (EBI) [7]. In addition to these there are several other databases consisting of DNA and/or protein sequence data along with their structural information. A major challenge today in biology is to make sense of the enormous amount of sequence data that is generated by large-scale genome sequencing projects. This explosion of biological sequence data, along with its high variability in acquisition and complex nature, warrants reliable and efficient computational techniques to augment the biologists' laborious wet laboratory techniques for interpreting the newly sequenced data. Hence, a new discipline, bioinformatics, which merges the current advances in molecular biology and computer algorithms has become increasingly important. The focus of this discipline is to make use of computer algorithms and sequence databases to analyze biological macromolecules/sequences, such as deoxyribonucleic acid (DNA), ribonucleic acid (RNA) and proteins, found in the cells of living organisms. The goal of bioinformatics is primarily to determine and analyze the complete collections of DNA (the genome) that comprises an organism.

An efficient analysis of biological sequences require the knowledge and collective

inputs from a diverse set of researchers such as biologists, statisticians and engineers. Computational methods have proved to be very promising for understanding biological sequences at the molecular level. The problem areas of mapping and sequencing, sequence analysis, structure prediction, phylogenic inference and regulatory analysis have been addressed successfully addressed using the techniques such as dynamic programming, Markov models, expectation-maximization, string search, clustering algorithms, etc. More recently, computer algorithms based on digital signal processing (DSP) techniques are becoming increasingly popular for analyzing and interpreting the features and functionality of DNA, RNA and protein sequences. Owing to the alphabetical nature of the biological sequences, which when mapped to numerical sequences, DSP techniques can be readily applied for their analysis. Powerful signal processing techniques, such as transform methods and digital filters, are now being successfully applied to address the research problems of predicting biologically significant features and structural information of genomic sequences. The results of these techniques have shown the need for further research in adapting the digital signal processing techniques to analyze and comprehend the complex nature of biological sequences. The following section gives a brief background and the progress in analysis of biological sequences using DSP techniques.

1.2 Background and a Brief Literature Review

Analysis of biological sequences involves identification of functionally significant patterns and is one of the main research problems in bioinformatics. It is well established that a great deal about the biological processes is better understood by studying these functionally significant patterns [8]. Some examples of such patterns are genes and CpG islands in DNA sequences, and hot-spots in proteins. Most of the biological sequences in order to stabilize conform into three dimensional (3D) structure and this

structural information of the biological sequences also aids in completely understanding their functionality.

In order to analyze DNA sequences, they need to be first extracted by liberating the cellular contents into a solution. The macromolecules in the solution such as DNA, RNA and proteins are then separated using either a centrifuge or chemical means (phenol extraction). Another technique of separating and purifying fragments of DNA or RNA as well as proteins is gel electrophoresis. The basic idea of electrophoresis is to separate the molecules based on their intrinsic electrical charge.

Once the DNA is isolated there are several experimental techniques available to analyze them [9]. For example, the experimental isolation of protein coding sequences (exons) in DNA is done using a method called *exon trapping*. This method relies on the fact that exons are flanked by splice recognition sites that are used during RNA processing to splice out the introns (non-coding sequences). The DNA containing the exons to be trapped is cut into segments using an appropriate restriction enzyme. Other experimental techniques for exon isolation are *radiation hybrid mapping* and the classical *genetic mapping*. The structural features of RNAs are of major importance to their biological functions such as coding, information transfer and catalytic activities. Proper functioning of RNAs require the formation of intricate three-dimensional (3D) structures. Protein sequences, like RNA, also fold into 3D structures and the knowledge of protein structure provides valuable information on the architecture and chemistry of a protein-protein interaction during biological processes. A technique known as x-ray crystallography has contributed to the determination of atomic-resolution large RNA and protein structures. The thermodynamics of protein-protein interactions can now be probed experimentally by a process called *alanine scanning mutagenesis*. It is now well established that only a small subset of contact residues in proteins contribute significantly to the binding free energy. These residues are known as "hot-spots" and if mutated they can disrupt the protein-protein

interaction. Although the above mentioned experimental techniques are very effective in producing accurate results, they involve several intricate time consuming steps that make the entire process laborious and expensive. Therefore, there is a strong need for computational techniques which are effective, reliable and economical for the analysis of biological sequences. The results obtained by the computational techniques can be a precursor for the biologists to base their experiments accordingly, and save time and resources.

There have been many computational methods [10–16] developed for solving the problem of CpG island (CGI) prediction in DNA sequences. These methods can be broadly categorized into two groups: (1) the traditional algorithms that are based on the three sequence parameters (length, C+G nucleotide content, ratio of the observed to expected CpG dinucleotides), and (2) the algorithms based on statistical properties in the DNA sequence. Most of the traditional methods, apart from identifying CGIs, have a tendency to falsely identify the other C and G rich motifs, e.g., *Alu repeats* as CGIs. In the subsequent methods the above three sequence parameters were made more stringent [17] in order to reduce false identification at the expense of missing some true CGIs. The statistical based methods [18,19] , which rely on the physical distance distribution of CpG dinucleoetides in a DNA sequence, have certain advantages, as they are not window based, but they suffer from low identification specificity.

In eukaryotic DNA, the genes are seperated by intergenic regions. The genes in turn has an alternating arrangement of protein coding (exons) and non-coding (introns) regions. This complex structure of genes poses a challenge in solving the problem of prediction of protein coding regions in eukaryotes. Most of the available gene finding methods, such as AUGUSTUS [20], GeneID [21], GenScan [22], HMMgene [23] and the methods developed by combining several gene-finding programs [24, 25], are data-driven. These methods involve performing a similarity search between a given unannotated sequence and annotated sequences from a database to

predict the properties of the former, and hence, are computationally expensive.

It is well established that an RNA sequence has a tendency to fold and twine about itself forming a stable three-dimensional (3D) structure [9]. Prediction of this stable RNA structure involves determining the locations of its sub-structures: stems, loops and pseudoknots. Most of the early computational methods for RNA secondary structure prediction were based on different heuristic search procedures minimizing the molecular energy of the RNA. Two such methods, involving quasi-Monte Carlo and genetic algorithm based search heuristics have been proposed in [26] and [27] respectively. A method based on maximum weighted matching (MWM) was proposed in [28], in which the possible base-pairs in RNA were determined by comparative sequence analysis. But this method is suitable only for RNA sequences for which the information on multiple alignments exist. Subsequently, an RNA secondary structure prediction algorithm called Mfold [29] was proposed, which is based on minimizing equilibrium free energy of RNA molecule using dynamic programming. There are other similar methods proposed [30–33] for RNA secondary structure prediction. Unfortunately, all the methods mentioned above fail to predict an accurate RNA secondary structure if it contains pseudoknots, which go undetermined. For determining pseudoknots in RNA secondary structure, dynamic programing based methods such as Pknots [34] and PknotsRG [35] have been developed. Several grammatical approaches [36–39] for RNA secondary structure prediction, which are based on multiple context-free grammar, have also been proposed which are capable of predicting pseudoknots, but suffer from computational complexity issues.

Similar to RNA sequences, protein sequences also possess a three-dimensional structure. Predicting protein secondary structure involves determining its substructures namely, α-helices, β-sheets, and loops. Early techniques, such as Chou-Fasman technique [40] and GOR [41], based on statistical characteristics of protein residues offered low prediction accuracies of 50-60%. In the late 1980's, for the first time, a

fully-connected multi-layer perceptron (MLP) neural network trained with backpropagation algorithm was used to achieve prediction accuracy of about 66% [42]. Later, a relatively successful technique, named as PHD [43], was developed exploiting evolutionary information contained in multiple sequence alignments of protein sequences. The PHD technique further increased the prediction accuracy to around 70% [44]. This was followed by the development of several techniques that combined evolutionary information of divergent proteins with neural networks [43, 45, 46]. Most of the existing protein structure prediction methods use a complicated scheme of input encoding to neural network prediction models in order to incorporate the evolutionary information. Moreover, the enormous growth of protein databases requires the existing prediction models to be extended using huge amounts of training data and developing large-scale neural networks.

The existing computational methods for hot-spot prediction [47, 48] in a protein sequence require complex structural information on its chemical composition, number of hydrogen bonds and binding free energy. This information is obtained using experimental techniques such as x-ray crystallography and alanine scanning. The dependence of computational methods, for hot-spot prediction, on experimental techniques slows down the entire prediction process. Moreover, the prediction of hot-spot locations in newly-discovered proteins becomes difficult as the detailed structural and physical information for these proteins is not yet available. Recently, techniques involving estimating the binding free energy using simulations of molecular dynamics has been proposed [49, 50] for prediction of hot-spots in proteins. Although, these methods produce encouraging results, they are difficult to implement mainly due to the complex models used for molecular dynamics.

The discrete nature of biological sequences has prompted many researchers to use digital signal processing (DSP) techniques for the analysis of biological sequences. The advent of sophisticated DSP techniques for analysis of biological sequences have

helped to alleviate the excessive cost and improve the accuracy of the computational methods. After mapping the biological sequences, which are alphabetical in nature, to appropriate numerical sequences, a number of DSP techniques [51–54] have been employed for biological sequence analysis.

Most of the DSP based techniques for predicting protein coding regions (exons) in DNA sequences exploit the period-3 property exhibited by exons. This period-3 property is due to a specific periodic arrangement of nucleotides in exons. By applying DSP techniques such as the sliding window DFT [55], digital filters [53, 56, 57], wavelet transform [58], and multirate DSP models [59, 60] researchers have successfully identified the locations of exons by detecting the period-3 segments in the DNA frequency spectrum. But most of the DSP techniques still fail to accurately locate short exons or the exons separated by short introns. Since in these cases, it is difficult to exactly locate the boundaries of exons. For predicting CGIs in DNA sequences, advanced methods [61, 62] utilizing two Markov chain models, one for CGIs and the other for non-CGIs, have been proposed. These two Markov models differ in their respective model parameters characterized by the transition probabilities between successive nucleotides. In these methods, a DNA segment is classified as CGI, if the value of a log-score computed using Markov model for a CGI is greater than that computed using Markov model for a non-CGI. More recently, CGI prediction methods utilizing digital filters [63, 64] have been proposed. These methods also make use of the Markov model parameters for identification of CGIs. The model parameters used for CGIs and non-CGIs play a crucial role in CGI identification. The use of different model parameters have sometimes produced contradicting results. A DSP based method [65] based on matched filtering has been proposed to predict the stem patterns in RNA secondary structure. The method significantly reduced the computational complexity but fails to predict pseudoknots in the RNA secondary structure. In recent years a number of DSP techniques, also for predicting hot-spots

in proteins, have been proposed. These techniques are based on using short-time discrete Fourier transform (STDFT) [66, 67] and modified Morlet continuous-wavelet transform [54]. These methods make use of the characteristic frequency of hot-spots. Unfortunately, these methods are not quiet reliable, as they tend to produce false positives.

From the above discussion, it is seen that the DSP techniques have come a long way in solving many important problems in the analysis of biological sequences. However, the performance of these techniques is still limited by the amount of the information on the characteristics of the biological sequences utilized as well as on the relevance of the DSP techniques employed. Hence, it is imperative to look into other relevant characteristics of biological sequences and incorporate these for the analysis of these sequences. At the same time, it could be useful to employ other more sophisticated DSP techniques for the analysis of these sequences with or without the use of additional information on their characteristics.

1.3 Scope of the Thesis

The objective of this research is to develop efficient and reliable digital signal processing (DSP) based techniques for the analysis of biological sequences. To this end, a number of challenging problems in the analysis of genomic and proteomic sequences are investigated in this thesis.

In the first part of the thesis, a study is undertaken to investigate the problem of identifying CpG islands and protein coding regions (exons) in DNA sequences. In this investigation, statistically optimal null filters (SONFs) are studied to effectively predict the locations of the characteristic properties pertaining to CpG islands and exons. Statistically optimal null filters are known for their ability to efficiently estimate short-duration signals embedded in noise and hence are expected to perform

efficiently in the above mentioned problems.

In the second part of the thesis, a systematic study is conducted for predicting the presence of pseudoknots in RNA sequences. Based on this study, matched filters are designed to determine the stem patterns in the dot-plot representation of RNA. These stem patterns in turn are expected to reveal the presence of pseudoknots.

Finally, in the third part of the thesis, the problems of predicting the secondary structure of proteins, and identifying hot-spots in proteins are investigated. Through this study a two-stage neural network based model is developed for accurately predicting the secondary structure of proteins. A scheme based on statistically optimal null filters is also developed for identifying hot-spots in protein sequences.

1.4 Organization of the Thesis

This thesis is organized as follows.

In Chapter 2, the background material necessary for the research work undertaken in this thesis is given. The chapter begins with a brief introduction to genomics describing the central dogma of molecular biology, which explains how proteins are synthesized from DNA. This is followed by description of various mappings of biological sequences to numerical sequences. The statistically optimal null filters, which are used extensively in this work, are also briefly reviewed. Finally, a brief account of various performance measures employed for the performance analysis of the proposed techniques is given.

In Chapter 3, DNA sequences are analyzed in order to investigate the two important research problems (1) identification of CpG islands and (2) prediction of protein coding regions (exons). An identification feature that characterizes a CGI is used to develop statistically optimal null filters (SONF) for the identification of CGIs in DNA sequences. The problem of predicting protein coding regions (exons) in DNA

sequences is also investigated using SONF. For this purpose, a basis function based on the well known period-3 property exhibited by exons is designed. The performance of each of the techniques developed is compared with the other existing state-of-the-art methods for the analysis of DNA sequences.

In Chapter 4, RNA sequences are studied to predict their secondary structure with pseudoknots. For this purpose, matched filters based on 2D convolution are developed to first identify the numbers and locations of stem and loop patterns in the RNA secondary structure. The knowledge of the stem and loop patterns are then used to deduce the presence of pseudoknot in an RNA structure. A graphical user interface (GUI) is also developed using MATLAB which displays the secondary structure of the RNA sequence.

In Chapter 5, protein sequences are investigated to solve the important problems of prediction of the protein secondary structure and prediction of the locations of hot-spots in proteins. The first problem is solved by developing a two-stage neural network scheme. The second problem of predicting the locations of hot-spots in proteins is solved using statistically optimal null filters. Hot-spots in proteins exhibit a characteristic frequency corresponding to their biological function. SONF is used to detect the locations of hot-spots belonging to a functional group by formulating basis functions having the characteristic frequency corresponding to that functional group.

Finally, Chapter 6, summarizes the study undertaken in this thesis and highlights its contributions. Some suggestions for further work based on the ideas and schemes developed in this thesis are also given.

Chapter 2

Background Material

2.1 Introduction

The Human Genome Project [5], which aims at sequencing and mapping of all the genes in humans, has garnered an immense interest in the scientific community. This project has resulted in large sets of genetic data and analyzing this data is of paramount importance. The recent statistical approaches for data analysis, signal processing techniques and control theory are well suited for this type of study. Consequently, the full potential of the area of genomics, which concerns the study of genomic data, can only be tapped by collective skills and creativity of a diverse set of researches including biologists, statisticians and engineers.

Before one proceeds with the analysis of biological sequences, it is essential to have some basic understanding of the molecular structures and the underlying cellular processes within these sequences. The objective of this chapter is to provide background in genomics and proteomics necessary for the analysis of biological sequences. Some discussions on numerical mappings of genomic and proteomic data, and statistically optimal null filters (SONF) are also provided in this chapter in view

of their importance in the DSP based approaches for the analysis of the biological sequences. Finally, some of the metrics to be used to evaluate the performance of the techniques for biological sequences are also briefly reviewed.

2.2 Biological Cells

The most fascinating thing about life is not its diversity but its fundamental building block. All living organisms are made up of microscopic fundamental biological structures called cells. Even though the cells are very tiny, each of them are in turn made up of complex cellular substructures. Each living cell generates its own energy and synthesizes its own macromolecules required for other biological processes. Some organisms such as bacteria and baker's yeast are unicellular, i.e., they contain only a single cell. Most other organisms are multicellular, containing many different type of cells. For example, the human body is composed of around 60 trillion cells with varying biological and structural properties.

Living cells may be divided into two types, the simpler prokaryotic cell and the more complex eukaryotic cell. By definition, prokaryotes are those organisms whose cells are not subdivided by membranes into a separate nucleus and cytoplasm. All prokaryote cell components are located together in the same compartment. In contrast, the larger and more complicated cells of higher organisms (animals, fungi and plants) are subdivided into separate compartments and are called eukaryotic cells. Figure 2.1 depicts the composition of prokaryotic and eukaryotic cells.

All living cells contain the essential chemical and structural components necessary for supporting life. For example, each bacterial cell has a single chromosome carrying a full set of genes providing it with the genetic information necessary to operate as a living organism. More complex organisms have genetic information much more than that of a bacteria. Humans have two duplicate sets of 23 different chromosomes,

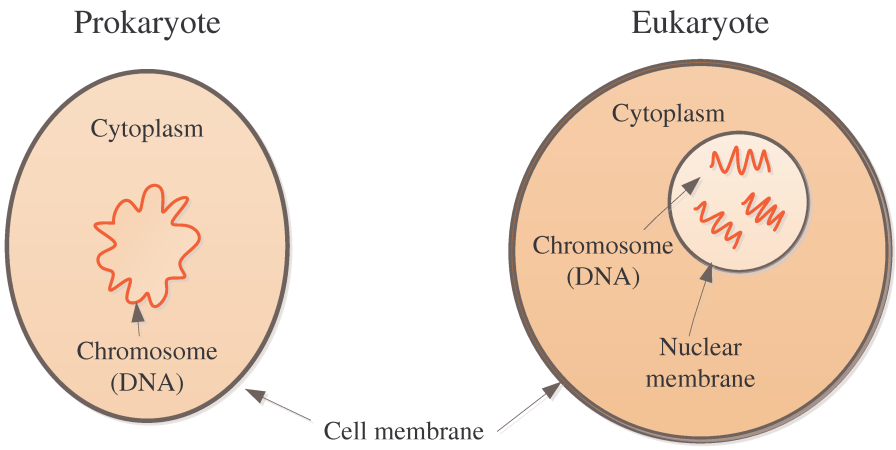


Figure 2.1: Difference between prokaryotic and eukaryotic cells.

making a total of 46 chromosomes. The complete set of chromosomes found in the cells of a particular individual is known as the karyotype [61].

2.2.1 DNA and RNA

Each chromosome is made up of a macromolecule, deoxyribonucleic acid (DNA), which is tightly coiled many times around proteins called histones, that support the structure of chromosome. Segments of DNA, called genes, contain the codes for genetic information. The word 'nucleic' in DNA arises from the fact that it was originally isolated from the nucleus of eukaryotic cells. DNA consists of subunits known as nucleotides, which are arranged linearly forming a DNA strand. DNA molecule has two such strands wound around each other in a helical arrangement. Figure 2.2(a) depicts the famous double helix first proposed by Francis Crick and James Watson in 1953 [1]. There are four different types of nucleotides associated with DNA, which are known as adenine, guanine, cytosine and thymine refered to as A, G, C and T, respectively. Figure 2.2(b) shows a more detailed and flattened segment of a DNA molecule. Each nucleotide in DNA has three components: a phosphate group, a five-carbon sugar, and a nitrogen-containing base. The phosphate groups and the sugars form the backbone of each strand of DNA. The bases are joined to the sugars and

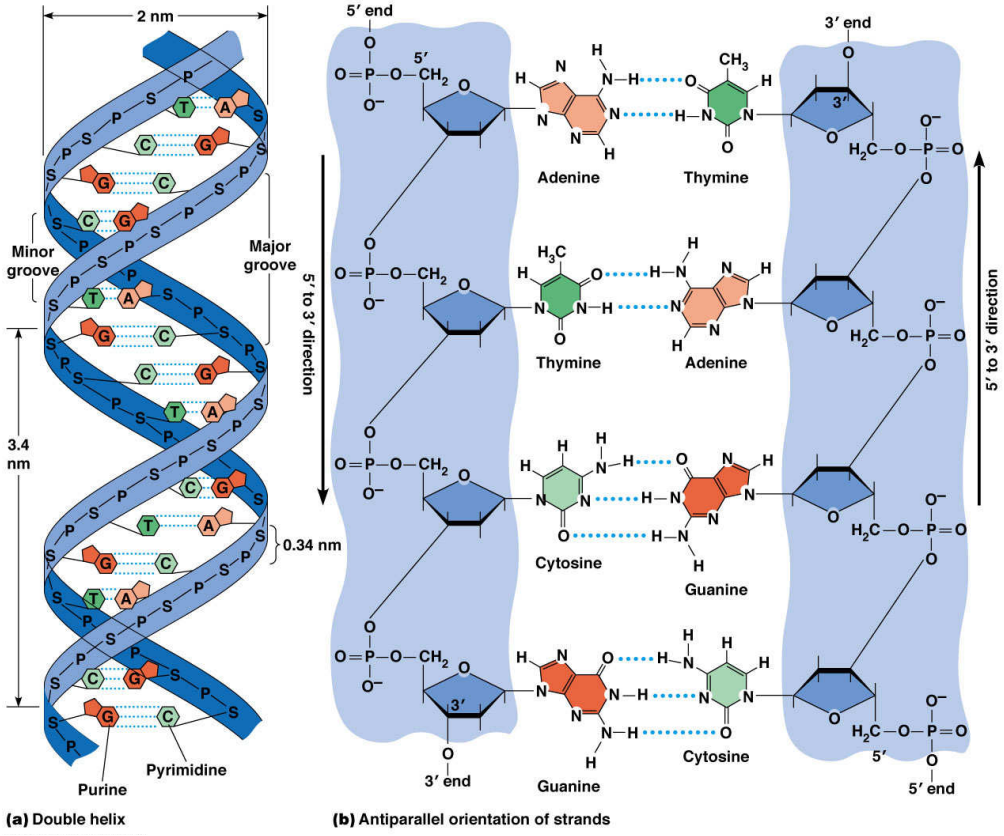


Figure 2.2: The DNA molecule. (a) DNA double helix and (b) Flattened DNA molecule. Source [68].

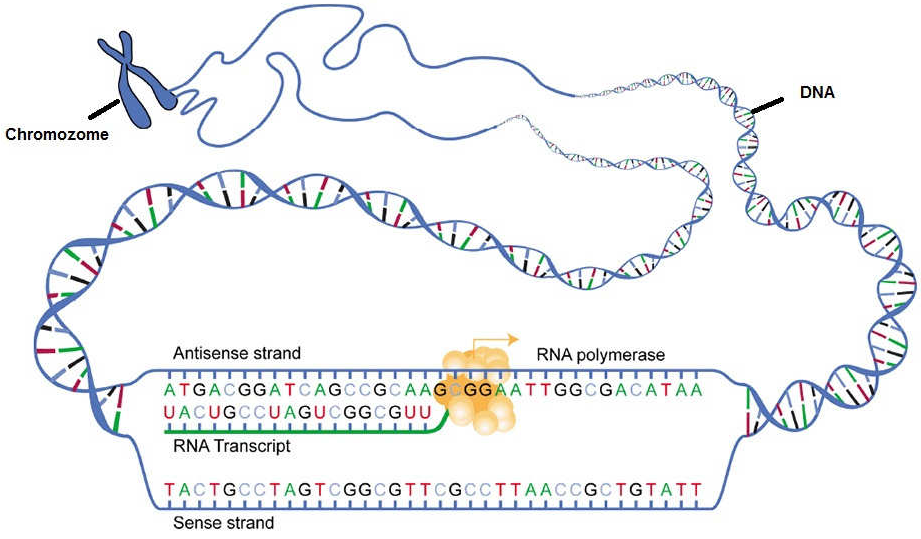


Figure 2.3: The chromosome, DNA molecule and the RNA transcript being created.

stick out sideways. Nucleotides are joined by linking the phosphate on the 5′-carbon of the (deoxy) ribose of one nucleotide to the 3′-position of the next as shown in the Figure 2.2. The phosphate group is joined to the sugar on either side by ester linkages known as a phosphodiester linkage. Conventionally, a strand of nucleic acids has direction and the 5′-end is regarded as the beginning of a DNA strand. The two strands of a DNA molecule are anti-parallel, as they point in opposite directions. This means that the 5′-end of one strand is opposite the 3′-end of the other strand. The DNA double helix is stabilized both by hydrogen bonds between the bases (Figure 2.2(a)) and by stacking of the aromatic rings of the bases towards inside the double-helix.

Ribonucleic acid (RNA) is a working copy of DNA resulting form a process known as transcription based on the information contained in DNA. RNA is very similar to DNA except that in RNA the nucleotide uracil (U) replaces thymine (T) in DNA, and RNA is normally found as a single-stranded molecule, whereas DNA is double stranded. From the viewpoint of genetic information, T in DNA and U in RNA are equivalent. The main job of RNA is to transfer the genetic information contained in DNA from nucleus to ribosome for the creation of proteins. This process prevents the DNA from having to leave the nucleus. This process keeps the DNA and genetic code protected from being corrupted. Figure 2.3 shows chromosome containing DNA molecule and the process of RNA transcription.

2.2.2 Proteins

In eukaryotic cells, the RNA created from the process of transcription leaves the nucleus and enters the cytoplasm as shown in Figure 2.4. The sequence of nucleotides in RNA are 'read' in groups of three by the ribosomes present in the cytoplasm, and translated into a chain of amino acids called protein. This process of synthesis of proteins using the genetic information coppied in RNA is called *translation*. In translation, a group of three consecutive nucleotides in RNA, referred to as a codon,

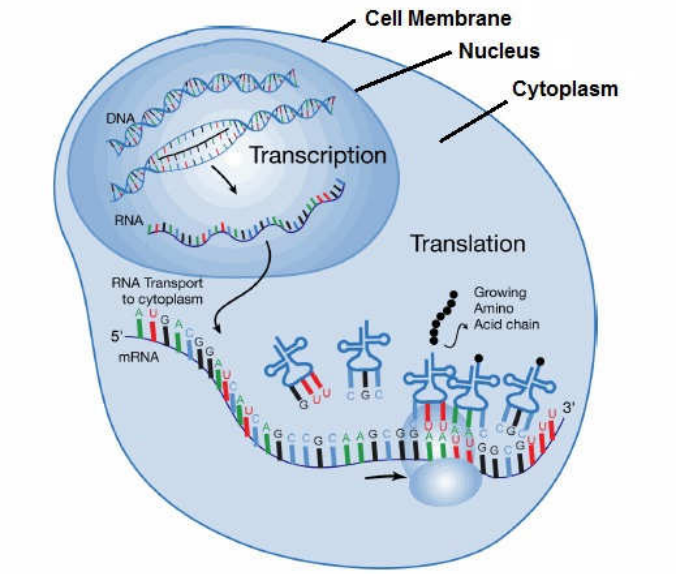


Figure 2.4: Transcription and translation of genetic information.

is responsible for the creation of a particular amino acid. Thus, the linear sequence of nucleotides in the RNA corresponds to the linear sequence of the amino acids that constitute a molecule of protein. The proteins are sometimes also reffered to as polypeptide chains as its constituent amino acids are joined by peptide bonds. There are 20 different amino acids, given in Table 2.1, which make up different kinds of proteins. Since there are four different nucleotides in RNA, there are 64 possible groups of three bases; i.e, 64 different codons in the genetic code (given in Table 2.2). As there are only 20 different amino acids, some of these are encoded by more than one codon. The codon AUG, in addition to encoding methionine, also acts as a start codon, which starts the process of translation. The three codons UAA, UAG and UGA are used for punctuation to stop the process of translation. Thus every new protein starts with the amino acid methionine (Met). An example of translation of RNA is shown in the Figure 2.5.

Proteins make up about two-thirds of the organic matters in a typical cell, and are directly responsible for most of the processes of metabolism. Proteins also perform most of the enzyme reactions such as catalyzing biochemical reactions, generating

Table 2.1: The Twenty Amino Acids

Leucine (L, Leu)	Tyrosine (Y, Tyr)
Isoleucine (I, Ile)	Tryptophan (W, Trp)
Asparagine (N, Asn)	Glutamine (Q, Gln)
Glycine (G, Gly)	Methionine (M, Met)
Valine (V, Val)	Serine (S, Ser)
Glutamic acid (E, Glu)	Cysteine (C, Cys)
Proline (P, Pro)	Threonine (T, Thr)
Histidine (H, His)	Phenylalanine (F, Phe)
Lysine (K, Lys)	Arginine (R, Arg)
Alanine (A, Ala)	Aspartic acid (D, Asp)

of energy, synthesizing of nucleotides, and transport functions of the cell such as transporting nutrients or taking part in cell movement. Generally, the molecules, such as proteins and most non-translated RNA, that form cellular structures or have active roles in carrying out reactions are normally folded into three-dimensional (3D) structures.

This scheme of transfer of genetic information from DNA to RNA and finally to protein as shown in the Figure 2.6 is known as central dogma of molecular biology.

2.3 Numerical Mapping of Biological Sequences

The biological sequences are alphabetical in nature, for example, the DNA sequences consists of an alphabet of four, and the protein sequences consists of an alphabet of twenty. Due to this reason these sequences need to be first mapped to numerical sequences in order to employ digital signal processing (DSP) based techniques for their

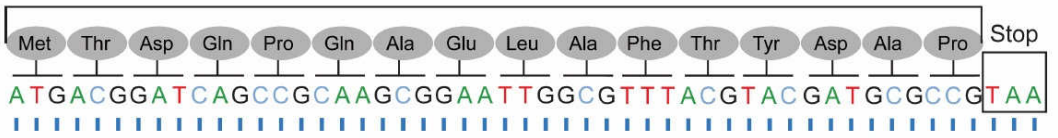


Figure 2.5: Translation of codons to amino acids.

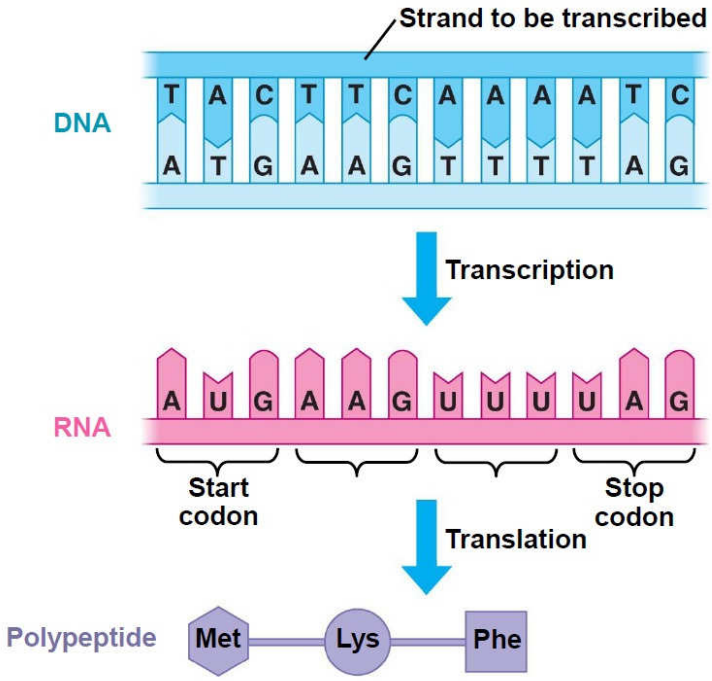
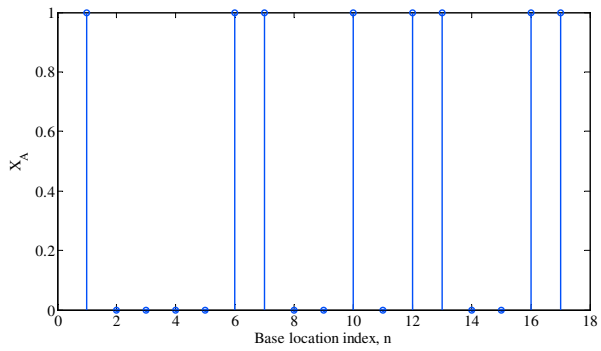


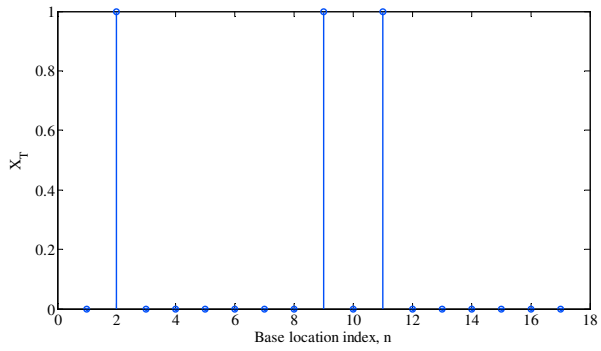
Figure 2.6: Central dogma of molecular biology. (Source [68])

analysis. There are several mapping techniques reported in the literature [69]. One of the earliest and a popular mapping is that of Voss's binary indicator sequences [70]. According to this mapping, a DNA sequence, X, can be mapped to a set of four digital signals, also called as binary indicator sequences, namely, X_A, X_T, X_G and X_C. In each of these binary indicator sequences, '1' represents the presence and '0' the absence of the corresponding nucleotide bases A, T, G and C in X. For instance, considering a DNA sequence $X = \{ATCCGAAGTATAACGAA\}$, the binary indicator sequence corresponding to G, i.e., X_G can be expressed as $X_G = \{0000100100000100\}$. Indicator sequences for the remaining three nucleotides can be represented in a similar fashion as shown in Figure 2.7.

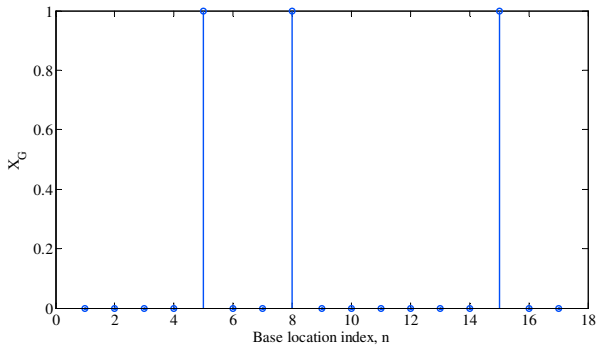
Another scheme of mapping is the one in which the electron-ion interaction potential (EIIP) values of the nucleotides are used to map the DNA sequence to a numerical sequence. EIIP values are physical quantities denoting average energy of valence electrons in the nucleotide bases [71]. Table 2.3 gives the EIIP values of the



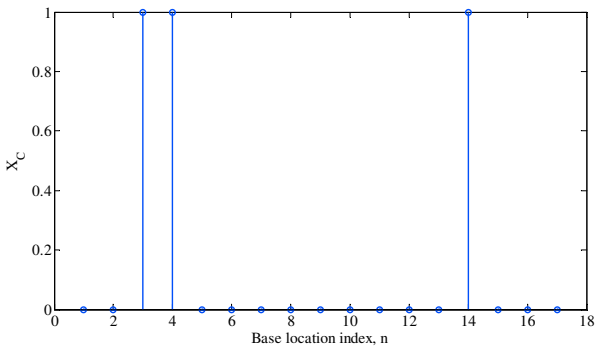
(a)



(b)



(c)



(d)

Figure 2.7: Binary indicator sequences. (a) Nucleotide A. (b) Nucleotide T. (c) Nucleotide G. (d) Nucleotide C.

Table 2.2: Genetic Code

AAA: K (Lys)	GAA: E (Glu)	UAA: STOP	CAA: Q (Gln)
AAG: K (Lys)	GAG: E (Glu)	UAG: STOP	CAG: Q (Gln)
AAU: N (Asn)	GAU D (Asp)	UAU: Y (Tyr)	CAU: H (His)
AAC: N (Asn)	GAC: D (Asp)	UAC: Y (Tyr)	CAC: H (His)
AGA: R (Arg)	GGA: G (Gly)	UGA: STOP	CGA:R (Arg)
AGG: R (Arg)	GGG: G (Gly)	UGG: W (Trp)	CGG: R (Arg)
AGU: S (Ser)	GGU: G (Gly)	UGU: C (Cys)	CGU: R (Arg)
AGC: S (Ser)	GGC: G (Gly)	UGC: C (Cys)	CGC: R (Arg)
AUA: I (Ile)	GUA: V (Val)	UUA: L (Leu)	CUA: L (Leu)
AUG: M (Met)/START	GUG: V (Val)	UUG: L (Leu)	CUG: L (Leu)
AUU: I (Ile)	GUU: V (Val)	UUU: F (Phe)	CUU: L (Leu)
AUC: I (Ile)	GUC: V (Val)	UUC: F (Phe)	CUC: L (Leu)
ACA: T (Thr)	GCA: A (Ala)	UCA: S (Ser)	CCA: P (Pro)
ACG: T (Thr)	GCG: A (Ala)	UCG: S (Ser)	CCG: P (Pro)
ACU: T (Thr)	GCU: A (Ala)	UCU: S (Ser)	CCU: P (Pro)
ACC: T (Thr)	GCC: A (Ala)	UCC: S (Ser)	CCC: P (Pro)

Table 2.3: EIIP Values for the Nucleotides in a DNA Sequence

Nucleotide	**EIIP Value**
Adenine (A)	0.1260
Thymine (T)	0.1335
Guanine (G)	0.0806
Cytosine (C)	0.1340

four nucleotides present in a DNA sequence. Figure 2.8 shows the EIIP sequence for the DNA sequence $X = \{ATCCGAAGTATAACGAA\}$.

EIIP-sequence can be interpreted as a weighted sum of the four indicator binary sequences as shown below where weights are the corresponding EIIP values.

$$X_{EIIP} = w_A X_A + w_T x_T + w_G X_G + w_C X_C \tag{2.1}$$

where w_A, w_T, w_G and w_C are the EIIP values as given in Table 2.3 and X_A, X_T, X_G and X_C are the respective indicator sequences. EIIP sequences involve only a single numerical sequence instead of four binary indicator sequences and hence this mapping

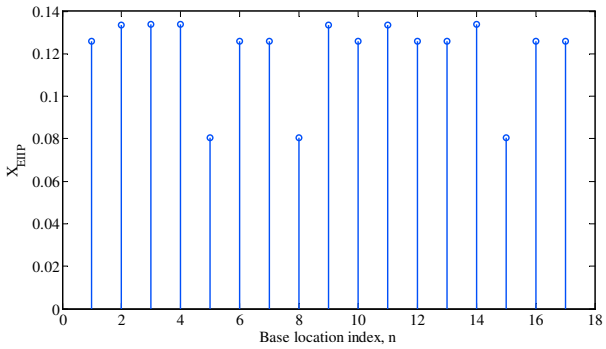


Figure 2.8: Mapping using EIIP values.

Table 2.4: EIIP Values of the Twenty Amino Acids in a Protein Sequence

Amino acid	**EIIP**	**Amino acid**	**EIIP**
Leucine (Leu)	0.0000	Tyrosine (Tyr)	0.0516
Isoleucine (Ile)	0.0000	Tryptophan (Trp)	0.0548
Asparagine (Asn)	0.0036	Glutamine (Gln)	0.0761
Glycine (Gly)	0.0050	Methionine (Met)	0.0823
Valine (Val)	0.0057	Serine (Ser)	0.0829
Glutamic acid (Glu)	0.0058	Cysteine (Cys)	0.0829
Proline (Pro)	0.0198	Threonine (Thr)	0.0941
Histidine (His)	0.0242	Phenylalanine (Phe)	0.0959
Lysine (Lys)	0.0371	Arginine (Arg)	0.0959
Alanine (Ala)	0.0373	Aspartic acid (Asp)	0.1263

more efficient in computational approaches.

Protein sequences can also be mapped numerical sequences using the EIIP values of its twenty amino acids. Table 2.4 gives the EIIP values of the twenty amino acids of a protein sequence.

2.4 Statistically Optimal Null Filters

In this section, a brief review of statistically optimal null filters (SONFs) [72], extensively used in this thesis, is given. Essentially, SONF is equivalent to a Kalman filter with a much simpler implementation and is able to effectively process short duration signals [72, 73]. Therefore, this property of SONF could be useful in identifying short

motifs in biological sequences. A brief description of SONF is now given in the context of processing genomic and proteomic sequences.

Consider a genomic or a proteomic sequence X, of length N. Now, the challenge is to identify the locations of occurrence of certain motifs such as CGIs, exons or hot-spots. In this work, statistically optimal null filters are utilized to perform this task as they are known for effective estimation of short duration signals embedded in noise. Here, the motifs are the short duration signals (or the message signals) and the residual signal is the noise. To be able to input the sequence, X, to SONF, it is first mapped to an appropriate numerical sequence. SONF being a window based approach, a sliding window of length L is used to determine whether a windowed numerical sequences, $X_n = \{x_n(m)\}$, where $n = 1, 2, \ldots, N-L+1$ and $m = n, n+1, \ldots, n+L-1$, belong to a particular motif or not. It can be noted that each of the windowed sequence, X_n, can be expressed as

$$X_n = S_n + R_n \tag{2.2}$$

where $S_n = \{s(m)\}$ is a message signal corresponding to the motif of interest and $R_n = \{r(m)\}$ is a residual signal. S_n and R_n are each of length L. SONF takes the windowed sequence, $X_n = \{x_n(m)\}$, as input and produces the output signal, Y_n, which is an optimal estimate of the message signal S_n. We define an SNR gain as the ratio of the variance of Y_n to the variance of X_n, given by

$$G(X_n) = \frac{\sum_{m=1}^{L}(y_n(m) - \bar{Y_n}^2)}{\sum_{m=1}^{L}(x_n(m) - \bar{X_n}^2)}, \tag{2.3}$$

where $\bar{Y_n} = (1/L)\sum_{m=1}^{L} y_n(m)$ and $\bar{X_n} = (1/L)\sum_{m=1}^{L} x_n(m)$. A windowed sequence can then be classified by comparing $G(X_n)$ with a prespecified threshold, η.

SONF produces the output Y_n by combining maximum signal-to-noise ratio and least squares optimization. The implementation of the two-fold optimization in SONF

approach is shown in Figure 2.9, where the instantaneous matched filter (IMF) [74] is first used to detect the presence of a short duration signal embedded in noise by maximizing the signal-to-noise ratio over variable-time observation interval, m.

The IMF output, I_n, is then scaled by a locally generated function, Λ_n, using least squares (LS) optimization procedure to obtain the optimal estimate, Y_n, of the message signal S_n. The key aspect of SONF is the formulation of a fixed binary basis sequence, $\Phi = \{\phi(m)\}$, of length equal to the size of the window, L. By modeling Φ according to some characteristic property of the motif, the message signal in the windowed sequence can be expressed as $S_n = V_n\Phi$, where $V_n = \{v(m)\}$ is also of length L. It is obvious that the sequence $V_n\Phi$ is obtained by multiplying together the corresponding elements of V_n and Φ. The SONF output, Y_n, is determined such that $Y_n \rightarrow S_n$ by minimizing the SONF output error (see Figure 2.9) in least square sense.

The following subsections explain in detail the steps involved in the SONF approach.

2.4.1 Instantaneous matched filter

The objective of instantaneous matched filter (IMF), which is the first stage of SONF (shown in Fig. 2.9), is to detect the presence of the waveform of the basis sequence, Φ, in the input sequence X_n. IMF is an improvement over a matched filter; the difference being, in IMF the optimal SNR is repeatedly calculated at every sample m, over an observation interval $m \in [n, n + L - 1]$. IMF takes X_n and Φ as inputs and produces an output sequence $I_n = \{\iota(m)\}$, where

$$\iota(m) = \sum_{i=n}^{m} x_n(i)\phi(i) \tag{2.4}$$

and $m = n, n + 1, \ldots, n + L - 1$. It can be seen that at each sample m, $\iota(m)$ is calculated over a varying interval $i \in [n, m]$.

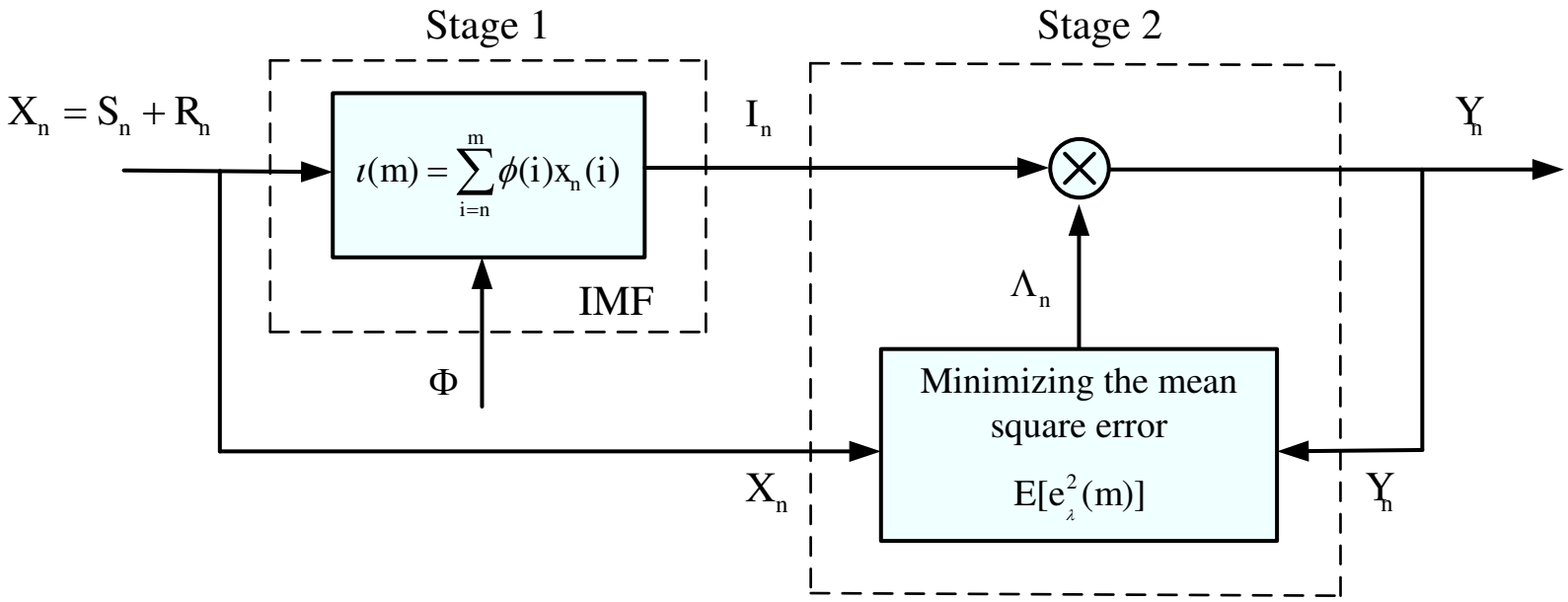


Figure 2.9: Statistically optimal null filter.

Note that, assuming $\iota(0) = 0$, $\iota(m)$ can also be calculated using the recursive relation given by

$$\iota(m) = \iota(m-1) + x_n(m)\phi(m). \tag{2.5}$$

The output $\iota(m)$ leads to an optimal detection of Φ at each sample m, and can be expressed as

$$\iota(m) = v(m)c(m) + r_0^{'}(m) \tag{2.6}$$

where, $v(m) \in V_n$ is an unknown gain, $r_0^{'}(m)$ is the residual signal in IMF output, and $c(m)$ is given by

$$c(m) = \sum_{i=n}^{m} \phi^2(i). \tag{2.7}$$

2.4.2 Least square optimization

The objective of the second stage of SONF is to determine a sequence $\Lambda_n = \{\lambda_n(m)\}$, which when used to scale the IMF output I_n, produces the SONF output, Y_n, such that $Y_n \to S_n$ in least square sense. Thus, $Y_n = \Lambda_n I_n$, is an estimate of S_n, which in turn, is an element wise product of V_n and Φ.

Let us consider the suboptimal case in which a sample of the IMF output $\iota(m)$ in (2.6), when scaled by $\lambda(m) = \phi(m)/c(m)$, yeilds

$$\begin{aligned} y(m) &= v(m)c(m) + r_0^{'}\frac{\phi(m)}{c(m)} \\ &= v(m)\phi(m) + r_0(m) \\ &= s(m) + r_0(m), \end{aligned} \tag{2.8}$$

where $y(m)$ is an element of the SONF output, Y_n. As we ideally desire $y(m) = s(m)$, the residual element, $r_0(m)$, needs to be entirely eliminated. Alternatively, the output error, $Z_n = \{z(m)\}$, given by

$$\begin{aligned} z(m) &= x(m) - y(m) \\ &= s(m) + r(m) - \lambda(m)\iota(m) \end{aligned} \tag{2.9}$$

should ideally be $z_{ideal}(m) = r(m)$.

The optimal Λ_n, $\Lambda_{opt} = \{\lambda_{opt}(m)\}$, is now determined by minimizing the mean square error, $E[e_\lambda^2(m)]$, with respect to $\lambda(m)$, where

$$\begin{aligned} e_\lambda(m) &= z_{ideal}(m) - z(m) \\ &= y(m) - s(m) \\ &= \lambda(m)\iota(m) - v(m)\phi(m). \end{aligned} \tag{2.10}$$

The optimal IMF scaling sequence $\lambda_{opt}(m)$ obtained by carrying out the above mean square minimization [72] is given by

$$\lambda_{opt}(m) = \frac{\phi(m)}{c(m) + 1/SNR} \tag{2.11}$$

where SNR is the input signal-to-noise ratio (considering $r(m)$ to be noise). According to this equation it is necessary to have the knowledge of input SNR in order to implement SONF. Since the input SNR is not readily available, a suboptimal case given by

$$\lambda_{subopt}(m) \rightarrow \frac{\phi(m)}{c(m)}. \tag{2.12}$$

is considered assuming $1/SNR << c(m)$. It can be shown that as m increases, $\lambda_{subopt}(m) \rightarrow \lambda_{opt}(m)$ since the second term in the right side of the equation

$$\frac{\lambda_{subopt}(m)}{\lambda_{opt}(m)} = 1 + \frac{1}{(SNR)c(m)} \tag{2.13}$$

approaches zero, since the value of $c(m)$ progressively increases as m increases. Thus, the value of input SNR in (2.11) will influence only the starting few samples in Y_n.

The SONF can be easily implemented by performing the steps given by the following set of equations [72]

$$\begin{aligned}
\iota(m) &= \iota(m-1) + x_n(m)\phi(m) && (2.14)\\
P(m) &= P(m-1) - \frac{P(m-1)\phi(m)\phi(m)P(m-1)}{1+\phi(m)P(m-1)\phi(m)} && (2.15)\\
\lambda(m) &= P(m)\phi(m) && (2.16)\\
y(m) &= \iota(m)\lambda(m). && (2.17)
\end{aligned}$$

In this work, the initial value of the gain $P(0)$ is chosen to be equal to 1, and it is assumed that $\iota(0) = 0$.

The block diagram of the SONF based analysis of genomic and proteomic sequences is given in Figure 2.10.

2.5 Performance Metrics

It is important to device appropriate performance metrics in order to effectively evaluate the performance of any algorithm. The objective of this thesis is to develop

algorithms for analyzing biological sequences, and such analysis involves predicting certain biologically important motifs in these sequences. Such prediction algorithms, in general, can have four possible outcomes; true positive (TP), true negative (TN), false positive (FP), or false negative (FN). A prediction is considered to be TP, if the prediction is truly positive in reality; otherwise, it is a considered to be FP. A prediction is considered to be TN, if the prediction is truly negative in reality; otherwise, it is considered to be a FN. These four prediction outcomes are shown in the Figure 2.11. The four prediction outcomes are determined by a classifier based on a specified threshold.

In this work, the performance of prediction algorithms is evaluated at the nucleotide level, i.e., the value of TP is obtained by adding all the nucleotides predicted to be true positive. The remaining three outcomes, FN, FP and TN, are calculated in the similar manner. Alternatively, at the motif level, even if one nucleotide (or a threshold of a minimum number of nucleotides) corresponding to a motif is predicted to be true positive, then the entire motif is assumed to be predicted correctly.

Sensitivity (Sn) and specificity (Sp) are the two basic metrics [75], which are commonly used to determine the accuracy of any prediction algorithm. Sensitivity is defined as the proportion of motifs that have been predicted correctly, and is given

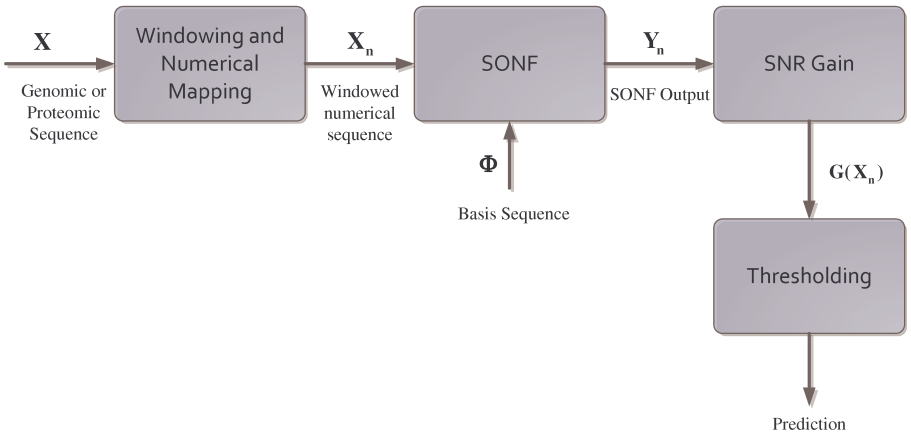


Figure 2.10: SONF based analysis of genomic and proteomic sequences.

by

$$Sn = \frac{TP}{TP + FN}. \tag{2.18}$$

Similarly, specificity is defined as the proportion of the predicted motifs that are true, and is given by

$$Sp = \frac{TP}{TP + FP}. \tag{2.19}$$

The values of both the sensitivity and specificity range from 0 to 1. For a perfect prediction, $Sn = 1$ and $Sp = 1$. Neither sensitivity nor specificity alone can provide a good measure of the global accuracy, since high sensitivity can be achieved with little specificity and vice versa. A metric that combines the values of sensitivity and specificity is called the Matthews correlation coefficient (CC), and is given by

$$CC = \frac{(TP \times TN) - (FN \times FP)}{\sqrt{(TP + FN)(TN + FP)(TP + FP)(TN + FN)}}. \tag{2.20}$$

The CC is in essence a measure of association between the actual and predicted locations of motif. The value of MCC ranges from -1 to 1, where a value of 1 corresponds to a perfect prediction; a value of -1 indicates that every positive location has been predicted as negative, and vice versa. Another important measure, called the performance accuracy (Acc), used in the performance evaluation of algorithms is

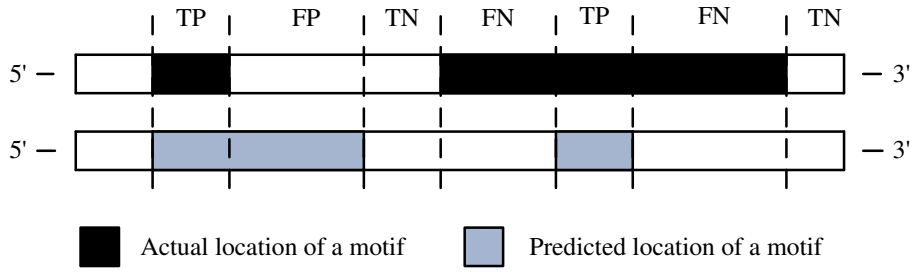


Figure 2.11: Four possible outcomes of a prediction algorithm.

given by

$$Acc = \frac{TP + TN}{TP + FP + TN + FN}. \tag{2.21}$$

Performance accuracy conveys the degree of closeness of predicted to the true locations of motif.

As mentioned, the four prediction outcomes, namely TP, TN, FP and FN, depend on the choice of the threshold parameter. The value of the threshold parameter chosen in turn would affect the performance metrics as they are determined using the four prediction outcomes. Hence, it is important to determine an optimal value of the threshold for accurate depiction of the performance metrics. For this purpose, a receiver operator characteristics (ROC) curve is generally utilized.

The ROC curves are obtained by plotting the true positive rate (TPR), which is same as Sn against the false positive rate (FPR), which is equal to $1 - Sp$, for different values of classification threshold parameter. An example of ROC curves for two classifiers C_1 and C_2 is given in Figure 2.12. The points on the curves are the values of TPR and FPR calculated for different thresholds. Since both TPR and FPR assume values in the range 0 to 1, the total area of the ROC plane is unity. The bottom left corner, (0, 0), represents the situation where the classifier predicts no positives. Another extreme situation of classification, represented by the top right corner, (1, 1), is when all instances are classified as positives. The top left corner, (0, 1), represents an ideal classifier with perfect classification with neither false positives nor false negatives. Finally, the bottom right corner, (1, 0), represents the worst classification with no true positives or true negatives.

The optimal threshold for a classifier is the one corresponding to the point on its ROC curve that is closest to the top-left corner. Additionally, by comparing the area under the ROC curves obtained for different prediction algorithms using the same dataset, the relative performance of these algorithms can also be examined. Greater

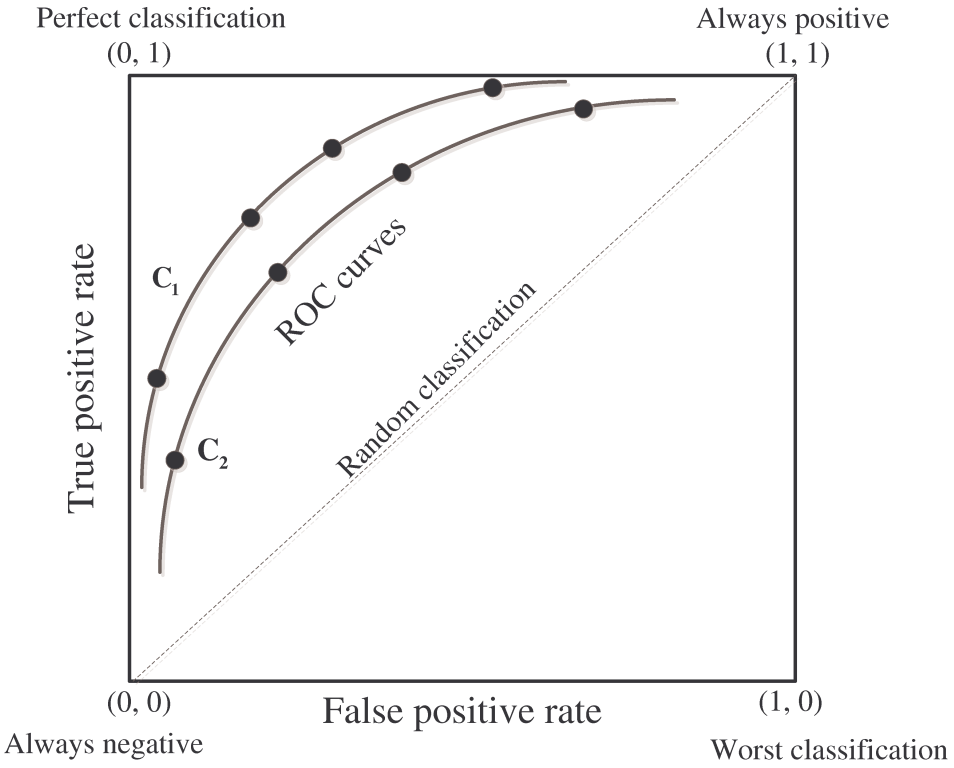


Figure 2.12: The receiver operating characteristic (ROC) curves.

the area under the curve, superior the performance of the prediction algorithm. In Figure 2.12, the ROC curve, C_1, corresponds to a prediction algorithm which is superior to that of C_2.

2.6 Summary

In this chapter, the background material necessary for the research work undertaken in this thesis has been presented. For the analysis of genomics and proteomics sequences, it is important to have some basic understanding of these macromolecules and the underlying cellular processes. Biological cells are the fundamental functional units in all living organisms. Simple organisms, such as most of the prokaryotes, are unicellular, and all its cellular constituents including the chromosome, which contains

the genetic information, are present in only one compartment enclosed by the cell membrane. Whereas, complex organisms, such as eukaryotes, are multicellular, and in each of these cells the chromosomes are enclosed in a nucleus, separated from other cellular constituents by a nuclear membrane. Chromosomes consists of the macromolecule, deoxyribonucleic acid (DNA), tightly wound around the protein called histone. Some segments of DNA, called the genes, carry genetic code in the form of a specific sequential arrangement of the nucleotides. The genetic code in genes is copied to ribonucleic acid (RNA) molecule by a process called transcription. This genetic code in RNA is then used by ribosomes present in the cytoplasm of the cell for the synthesis of proteins by a process called translation. Proteins perform a vast array of functions in living organisms, including catalyzing metabolic reactions, replicating DNA, responding to stimuli and many transport functions.

Biological sequences are alphabetical in nature and need to be mapped to numerical sequences so that many numerical techniques can be developed for biological sequence analysis. Hence, Two such mappings, the Voss's binary indicator mapping and the one using EIIP values of the nucleotides in biological sequences, have been briefly described. The statistically optimal null filters (SONFs), which are used extensively for the analysis of biological sequences in this thesis, have also been reviewed in this chapter. SONFs have the ability to track rapidly changing signals by combining maximum signal-to-noise ratio and least squares optimization criteria, leading to more practical processing of short-duration signals. Finally, a brief account of various performance metrics, such as sensitivity, specificity, correlation coefficient, performance accuracy and the ROC technique, which are extensively used for evaluation of the algorithms developed in this thesis, is given.

Chapter 3

Analysis of DNA Sequences

3.1 Introduction

Chromosomes in the nucleus of eukaryotic cells are made up of DNA containing about three billion base pairs. DNA contains around twenty five to thirty thousand genes with an average length of about three thousand base pairs. Identifying the locations of these genes is one of the important problems in the analysis of DNA sequences. As only about one percent of the DNA contains genes, which get transcribed into RNA, identifying the locations of genes is a huge challenge.

This problem of identifying the locations of genes in DNA can be dealt with by first finding the regions called promoters, which immediately precede the genes. The promoter regions are the binding sites for enzymes which perform the process of transcription of genes, thus serving as transcription start sites. Similarly, the regions which immediately follow the genes called, terminators, serve as transcription stop sites. The entire unit, shown in the Figure 3.1, comprising the gene along with the promoter and the terminator is called transcription unit. As every gene is preceded by a promoter, finding the promoter regions helps us in determining the locations of

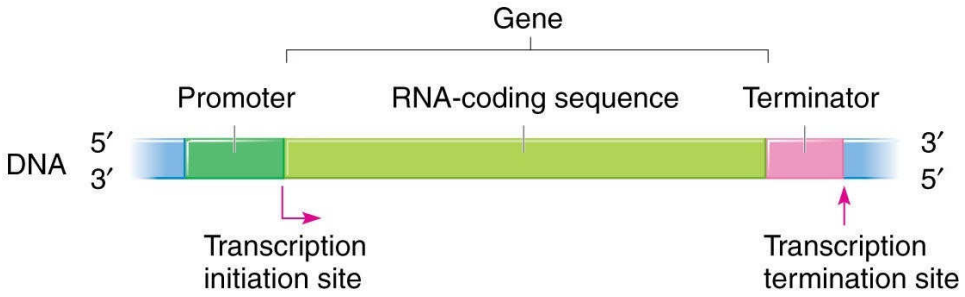


Figure 3.1: A transcription unit.

genes in DNA. The promoter regions contain CpG islands, which are the sequences dominated by the presence of CG dinucleotides [76]. The promoter regions can be identified by detecting such islands. These CpG islands have a certain characteristic property, which can be readily modeled for their identification using computational approaches.

CGIs, apart from playing an important role in promoter prediction, and consequently in the prediction of genes [77, 78], they also help promoters regulate the functionality of genes [79–82]. The CGIs in the promoter regions can be either methylated or unmethylated. Methylation of CGIs is a biochemical modification resulting from addition of a methyl group to the nucleotide cytosine (C). The unmethylated condition of CGIs help the promoters to regulate the genes they control by turning their gene expression 'ON'. On the contrary, the methylated condition of CGIs turn

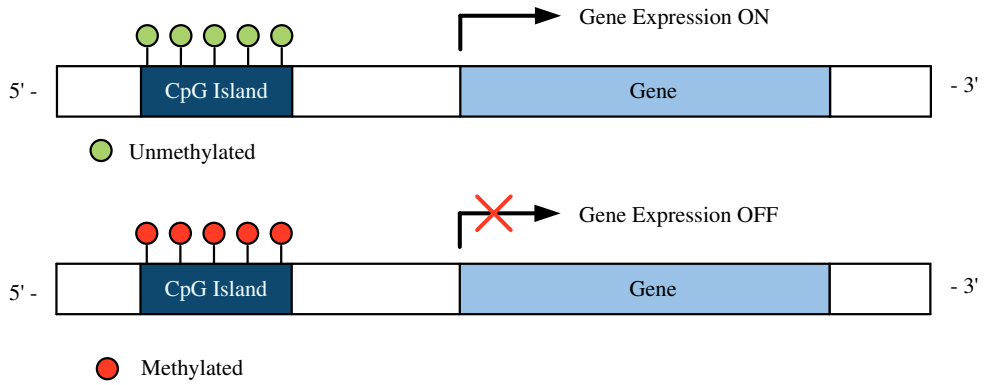


Figure 3.2: Difference between mythelated and unmythelated CpG island.

'OFF' the genes leading to gene silencing as shown in Figure 3.2. For example, if the CGIs belonging to promoters that regulate certain tumor suppressor genes are methylated, then the corresponding cells are prone to cancer. Thus, the methylated or unmethylated condition of CGIs can be used for early detection of deceases such as cancer [83–87]. Due to these reasons, identification of CGIs in DNA sequences has become indispensable for genome analysis and annotation.

The genes in eukaryotic DNA have an alternating arrangement of exons and introns. Exons and introns are respectively the protein coding and the non-coding regions of a gene. During the process of transcription the exons and introns in a gene are first transcribed to an initial RNA transcript, to which a *cap* and a *tail* are added as shown in the Figure 3.3. This allows the ribosome in the cytoplasm to recognize the RNA during translation. Before the RNA enters the cytoplasm, the segments of transcript corresponding to introns are removed resulting in a transcript containing only exons. The resulting RNA, called the messenger RNA (mRNA), is used for the synthesis of proteins. Therefore, it is necessary to predict the locations of exons in a gene, so that, the resulting protein sequence synthesized from mRNA can be accurately determined. The identification of exons has helped genetic engineers to isolate proteins performing the desired biological functions and has resulted in designing customized drugs for curing various diseases. Due to these reasons, the prediction of exons in DNA sequences is an important step in tackling the larger task of understanding biological processes. However, the alternating arrangement of exons and introns, with their varying lengths, poses a challenge in solving the problem of prediction of locations of exons in eukaryotic DNA.

In this chapter, the two above-mentioned important problems, namely, identification of CpG islands (CGIs) and prediction of protein coding regions (exons), in DNA sequences, are investigated [88–90]. Both these problems are investigated using statistically optimal null filters (SONFs), reviewed in Chapter 2. The characteristic

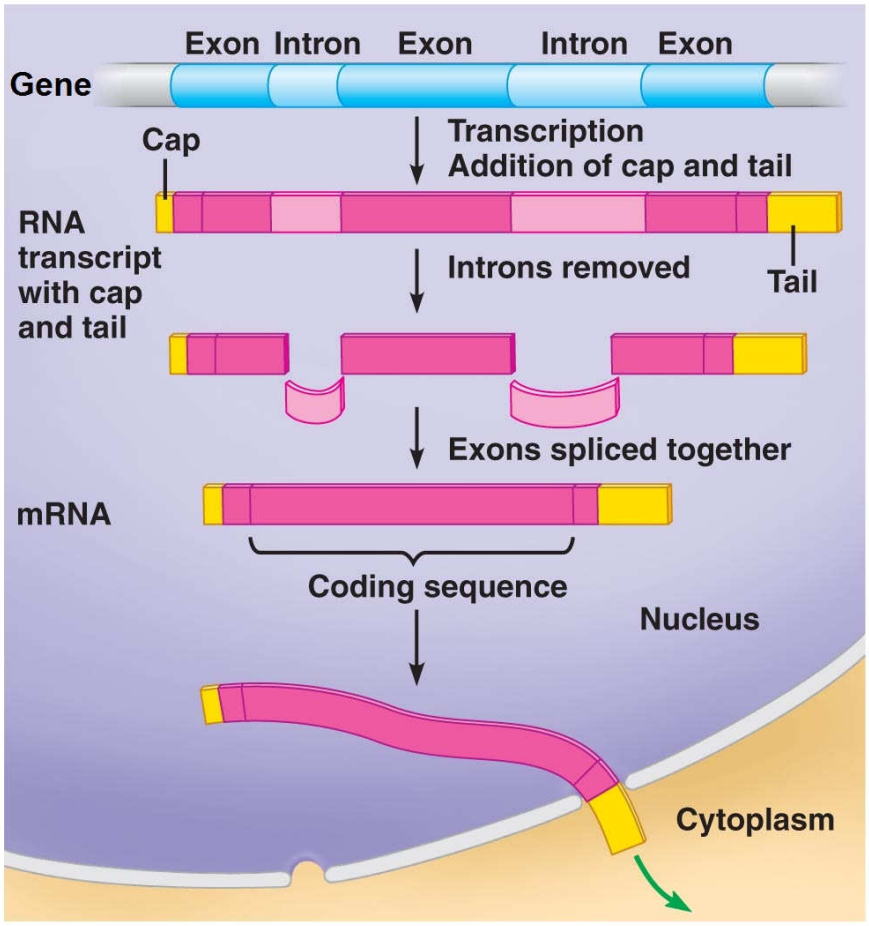


Figure 3.3: A gene containing exons and introns (Source [68]).

properties of CGIs and exons are individually modeled as the basis functions to be utilized by SONFs for solving the above two problems.

3.2 Identification of CGIs

A typical CpG island (CGI) in a DNA sequence consists of a high-frequency CpG dinucleoetides. CGIs vary in length from a few hundred to a few thousand base pairs (bp), but rarely exceeding 5000 bp. The 'p' in CpG refers to the phosphodiester bond between the adjacent C and G nucleotides of a DNA strand [61, 62]. This bond is different from the hydrogen bond that exists between C and G across two strands

in a DNA double helix. Formally, a CGI is defined as a DNA segment fulfilling the following three conditions: (i) length of segment is at least 200 bp, (ii) G and C content is $\geq$ 50%, and (iii) observed CpG to expected CpG ratio (o/e) is $\geq$ 0.6. Observed CpG is the number of CpG dinucleoetides in a segment and expected CpG is calculated by multiplying the number of 'C's and the number of 'G's in a segment and then dividing the product by the length of the segment.

The following section gives a brief review of some of the existing DSP based techniques for identification of CGIs.

3.2.1 Previous work

In this section a brief review of some of the existing CGI identification methods is given, which will be used for comparing the the performance of the method to be proposed in Section 3.2.2.

Markov chain approach

In this method, a DNA sequence $X = \{x(1), x(2), \ldots, x(n), \ldots, x(N)\}$ of length N, where each symbol $x(n) \in \{A, C, T, G\}$, is considered to be a first-order Markov chain [91]. This is due to the conditional independence property of X, i.e., the nucleotide occurring at the location $(n-1)$ does not provide any information over and above that at n in order to predict the nucleotide occurring at $(n+1)$. In a CpG island, the probability of transition from the nucleotide C to the nucleotide G is higher in comparison with that in a non-CGI. Let the probability of transition from a nucleotide β to a nucleotide γ in a CGI and a non-CGI be denoted as $p^{+}_{\beta\gamma}$ and $p^{-}_{\beta\gamma}$ respectively. Table 3.1 and Table 3.2 taken from [62], show the transition probabilities for CGI and non-CGI Markov models. These tables are derived from 48 putative CGIs and non-CGIs in human DNA sequences. Each row in these tables contains transition

Table 3.1: Transition Probabilities Inside a CGI [62]

$p^{+}_{\beta\gamma}$	**A**	**C**	**G**	**T**
A	0.180	0.274	0.426	0.120
C	0.171	0.368	**0.274**	0.188
G	0.161	0.339	0.375	0.125
T	0.079	0.355	0.384	0.182

probabilities from a specific nucleotide base to each of the four bases. These transition probabilities $p^{\pm}_{\beta\gamma}$ are calculated using

$$p^{\pm}_{\beta\gamma} = \frac{n^{\pm}_{\beta\gamma}}{\sum\limits_{k\in\{A,T,G,C\}} n^{\pm}_{\beta k}}, \tag{3.1}$$

where $n^{\pm}_{\beta\gamma}$ is the number of dinucleoetides $\beta\gamma$ in a DNA sequence. Naturally, every row in these tables adds up to unity. As expected, in Table 3.1, which corresponds to the CGI Markov model, the probability that a C is followed by a G is very high as compared with that in Table 3.2.

The CGIs, in the DNA sequence X, are identified by analyzing each of the windowed sequences of length L, $X_n = \{x(n), x(n+1), \ldots, x(n+L-1)\}$, which are obtained by shifting the window by one position at a time. The probability of observing a windowed sequence, X_n, assuming that it belongs to a CGI is given by

Table 3.2: Transition Probabilities Inside a Non-CGI [62]

$p^{-}_{\beta\gamma}$	**A**	**C**	**G**	**T**
A	0.300	0.205	0.285	0.210
C	0.322	0.298	**0.078**	0.302
G	0.248	0.246	0.298	0.208
T	0.177	0.239	0.292	0.292

$$
\begin{aligned}
&P(X_n|\text{CGI}) \\
&= P(x(n)\ldots x(n+L-1)|x(n-1), \text{CGI model}) \\
&= \prod_{i=0}^{L-1} p^{+}_{x(n-1+i)x(n+i)}.
\end{aligned} \tag{3.2}
$$

Similarly, the probability of observing, X_n, assuming it belongs to a non-CpG island region is

$$
\begin{aligned}
&P(X_n|\text{non-CGI}) \\
&= P(x(n)\ldots x(n+L-1)|x(n-1), \text{non-CGI}) \\
&= \prod_{i=0}^{L-1} p^{-}_{x(n-1+i)x(n+i)}
\end{aligned} \tag{3.3}
$$

If $P(X_n|\text{CGI}) > \text{P}(\text{X}_\text{n}|\text{non-CGI})$, then, it is concluded that the sequence X_n belongs to a CGI. Otherwise, it is considered to be a non-CGI. A CGI can also be identified by formulating a log-likelihood ratio, given by

$$
S(n) = \frac{1}{L}\log\frac{\text{P}(\text{X}_\text{n}|\text{CGI})}{\text{P}(\text{X}_\text{n}|\text{non-CGI})}. \tag{3.4}
$$

If $S(n) > 0$, the given DNA sequence is considered to belong to a CGI, and if $S(n) < 0$ the sequence is considered to be a non-CGI.

IIR low-pass filter approach

Byung-Jun Yoon *et al.* [63], have noted that the log-likelihood ratio given in (3.4) can be expressed as

$$
\begin{aligned}
S(n) &= \frac{1}{L}\log\prod_{\mathrm{n=0}}^{\mathrm{L-1}}\frac{\mathrm{p}^{+}_{\mathrm{x(n-1)x(n)}}}{\mathrm{p}^{-}_{\mathrm{x(n-1)x(n)}}} \\
&= \frac{1}{L}\sum_{i=0}^{L-1} y(n+i) \\
&= y(n) * h_{ave}(n), \qquad (3.5)
\end{aligned}
$$

where $y(n)$ is a sequence representing the log-likelihood ratio of a single transition given by

$$
y(n) = \log\left(\frac{\mathrm{p}^{+}_{\mathrm{x(n-1)x(n)}}}{\mathrm{p}^{-}_{\mathrm{x(n-1)x(n)}}}\right) \qquad (3.6)
$$

and, $h_{ave}(n)$ is a simple averaging filter defined as

$$
h_{ave}(n) = \begin{cases} 1/L, & \text{for } -L+1 \le n \le 0 \\ 0, & \text{otherwise.} \end{cases} \qquad (3.7)
$$

Then, they proposed using a bank of M filters, each having different bandwidth, instead of using simply one low pass filter $h_{ave}(n)$. Specifically, the filter used in the k^{th} $(k = 0, \ldots, M-1)$ channel has a transfer function given by

$$
H_k(z) = \frac{1-\alpha_k}{1-\alpha_k z^{-1}}, \qquad (3.8)
$$

where $0 < \alpha_0 < \alpha_1 < \cdots < \alpha_{M-1} < 1$. Since impulse response of a filter in the bank is $h_{ave}(k) = (1-\alpha_k)\alpha_k^k u(n)$, more recent inputs are given larger weights, than that to those preceding them, in the averaging process of $y(n)$. The filter bank consists of forty channels ($M = 40$), and the filter parameter α_k is chosen from 0.95 to 0.99 with an increment of 0.001. The log-likelihood ratio obtained from the output of the

k^{th} channel is given by

$$S_k(n) = y(n) * h_k(n). \tag{3.9}$$

The values of $S_k(n)$ obtained for all k and n are then used to obtain a two-level contour plot. The bands corresponding to $S_k(n) > 0$ determine the locations of CGIs.

In this method, the computational overhead increases considerably as the number of channels in the filter bank is increased.

Multinomial statistical model

This method by Ahmad Rushdi *et al.* [64], differs from the previous method by the way the transition tables are obtained and the type of digital filter used to calculate the log-likelihood ratio. Instead of using (3.1) to obtain the transition probability tables, they are generated using a multinomial model [92]. Transition probabilities, $p^{\pm}_{\beta\gamma}$ for the windowed sequence, X_n, are calculated as

$$p^{\pm}_{\beta\gamma} = \frac{c^{\pm}_{\beta\gamma}}{\sum\limits_{k\in\{A,T,G,C\}} c^{\pm}_{\beta k}} \tag{3.10}$$

where

$$c^{\pm}_{\beta\gamma} = \frac{frequency^{\pm}_{\beta\gamma}}{(frequency^{\pm}_{\beta})(frequency^{\pm}_{\gamma})}, \tag{3.11}$$

the symbols $frequency^{\pm}_{\beta\gamma}$ and $frequency^{\pm}_{\beta}$ representing, respectively, the frequency of occurrence of the dinucleotide $\beta\gamma$ and that of the nucleotide β. This method uses an FIR digital filter with variable coefficients generated using the Blackman window to calculate the log-likelihood ratio, $S(n)$. The locations having $S(n)$ greater than zero are considered to be CGIs.

All the above mentioned methods rely on the transition probability tables to

calculate log-likelihood ratio used to identify CGIs. The methods given in [63, 64] vary specifically by the way the weighting function is used to average $y(n)$, which is obtained from the respective transition tables. It is shown later in Section 3.3 that the choice of the transition tables may produce contrasting results. Hence, a more reliable and efficient scheme, that does not depend on the transition tables, is necessary for identifying CGIs.

3.2.2 Proposed SONF based method

In this work, the use of statistically optimally null filters (SONFs) is proposed [88, 89] to solve the problem of CGI identification in DNA sequences.

Consider an unannotated DNA sequence X, of length N, in which the locations of CGIs need to be identified. As mentioned in Section 2.4, SONFs are suitable to perform this task as they are known for effective estimation of short duration signals embedded in noise. Here, the CGIs are the short duration signals (or the message signals) in the DNA sequence X, and the residual signal is the noise. To be able to feed the sequence X to SONF, it is first mapped to an appropriate numerical sequence $X_{CG} = \{x_{CG}(n)\}$. SONF being a window based approach, a sliding window of length L is used to determine whether or not a windowed sequences of X_{CG}, $X_n = \{x_n(m)\}$, where $n = 1, 2, \ldots, N - L + 1$ and $m = n, n + 1, \ldots, n + L - 1$, belong to a CGI. It can be noted that each windowed sequence, X_n, can be expressed as

$$X_n = S_n + R_n, \tag{3.12}$$

where $S_n = \{s(m)\}$ is a message signal corresponding to the CGI, and $R_n = \{r(m)\}$ is a residual signal. S_n and R_n are each of length L. SONF takes the windowed sequence, $X_n = \{x_n(m)\}$ and a basis sequence, $\Phi = \{\phi(m)\}$, having some characteristic property of CGI, as inputs and produces the output signal, Y_n, which is an optimal estimate of

the message signal S_n. Now, by formulating an appropriate threshold on SNR gain, $G(X_n)$, which is the ratio of variance of Y_n to the variance of X_n, each of the windowed sequences can be classified as belonging to a CGI or not.

In the following, we will now describe in detail the numerical mapping of DNA sequences, formulation of the basis sequence based on some characteristic properties of CGIs, choice of the window length, and finally the algorithm for CGI identification.

Numerical mapping of DNA sequences

In Section 2.3, it was shown that a DNA sequence, X, can be mapped to a set of four digital signals by forming four binary indicator sequences, namely, X_A, X_T, X_G and X_C. In each of these binary indicator sequences, '1' represent the presence and '0' the absence of the corresponding bases A, T, G and C in X. For instance, considering a DNA sequence $X = \{ATCCGAAGTATAACGAA\}$, the binary indicator sequence corresponding to G, i.e., X_G can be expressed as $X_G = \{0000100100000100\}$. Indicator sequences for the remaining three nucleotides can be represented in a similar fashion.

The problem of CGI identification deals with G and C content in a DNA sequence. Hence, we define a new indicator sequence $X_{CG} = \{x_{CG}(n)\}$, which indicates the presence of both the nucleotides C and G in a DNA sequence. For example, the binary indicator sequence X_{CG} of the DNA sequence X given above is $X_{CG} = \{0011100100000100\}$.

Formulation of the basis sequence

A formulation of basis sequences, based on some characteristic properties of CGIs, is very important for identifying CGIs in the input DNA sequence. For this purpose, the CGIs in sequence L44140 [5] taken from the chromosome X of *Homo sapiens* are studied for some characteristic property. This sequence is of length 219447 bp and has

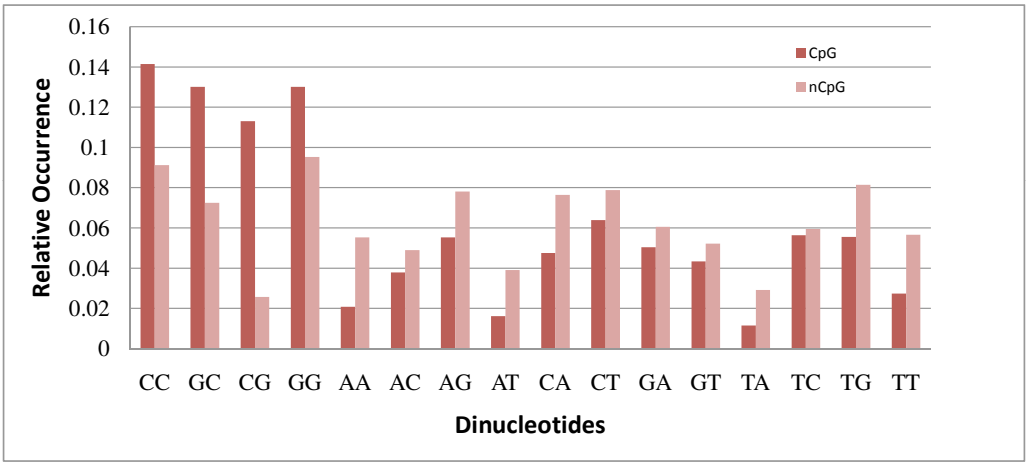


Figure 3.4: Comparison of relative occurrence of dinucleotides in CGIs and non-CGIs of L44140.

17 CGIs, whose locations have been reported in the NCBI website [5]. We calculate the occurrences of all the possible dinucleotides in the CGIs and the non-CGIs of the sequence L44140. The bar chart in Figure 3.4 depicts the relative occurrences of the dinucleotides in this sequence. Here, the relative occurrence of a particular dinucleotide is defined as the ratio of the number of times the dinucleotide occurs in the sequence to the sequence length. In the barchart of Figure 3.4, the darker bars corresponding to the dinucleotides CC, CG, GC, and GG are taller in CGIs; whereas, the darker bars corresponding to the other dinucleotides (AA, AC, AG, AT, CA, CT, GA, GT, TC, TG, TT, and TA) are shorter. It is evident from Figure 3.4 that the dinucleotides CC, CG, GC, and GG occur more frequently in CGIs, whereas the other dinucleotides occur more frequently in non-CGIs. Hence, it would be appropriate to consider the relative occurrences of the four dinucleotides CC, CG, GC, and GG, instead of only CG, in order to distinguish between a CGI and non-CGI.

Next we study the difference in gap sizes between the dinucleotides CC, CG, GC, and GG in CGIs and non-CGIs of the sequence L44140. The gap size between a pair of two consecutive dinucleotides, from these set of four, is defined as the number of nucleotides occurring between the pair. Figure 3.5 shows the relative occurrence of gaps of various sizes in a CGI and that in a non-CGI in this sequence. Here,

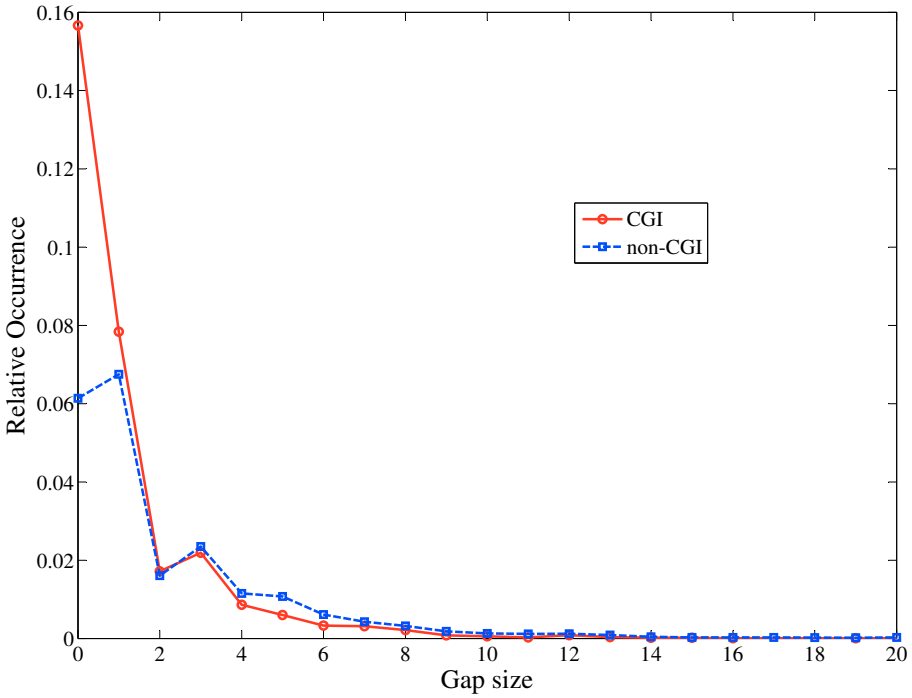


Figure 3.5: Relative occurrences of various gap sizes in CGIs and non-CGIs of L44140.

the relative occurrence of a particular gap size is defined as the ratio of the number of times the gap size occurs in the sequence to the sequence length. It is seen in Figure 3.5 that, the gap of size 0 occurs more frequently in a CGI as compared to that in a non-CGI. It is found that the gap size in a non-CGI can go up to 40 where as in CGIs the maximum gap size was found to be 19. It is also seen from Figure 3.5 that the gaps of sizes 0, 1 or 2 occur more frequently in a CGI, and the gap sizes of 3 or larger occur more frequently in a non-CGI. This observation, coupled with the fact that in a CGI atleast 50% of the nucleotide content is due to C and G, favors a formulation of the basis sequence as

$$\Phi = \{1100110011 \ldots 001100\}. \tag{3.13}$$

A '11' in the basis sequence Φ is meant to capture the presence of one of the four dinucleotides CC, CG, GC, and GG; whereas, a '00' is meant to capture the presence

of one of the remaining dinucleotides in a window of DNA sequence.

Window size

Now, in order to obtain the length of Φ (window size), we analyze CGIs and non-CGIs of different lengths for the relative occurrences of various gap sizes. Figure 3.6 shows the plot of Δ, the difference of relative occurrence of a particular gap in a CGI and a non-CGI, *versus* the window size for various gap sizes.

It can be seen from Figure 3.6 that, as the window size increases, Δ also increases before it reaches a steady value. Irrespective of the gap size considered, Δ stabilizes for window sizes greater than 200. As the number of computations increases with increasing size of the window, in the proposed method a window size of 200 is chosen.

It can be also seen from Figure 3.6 that Δ is maximum for gap size 0. The value of Δ is negative for gap size 3, signifying that the gap sizes of 3 or larger are more probable in non-CGIs than in CGIs.

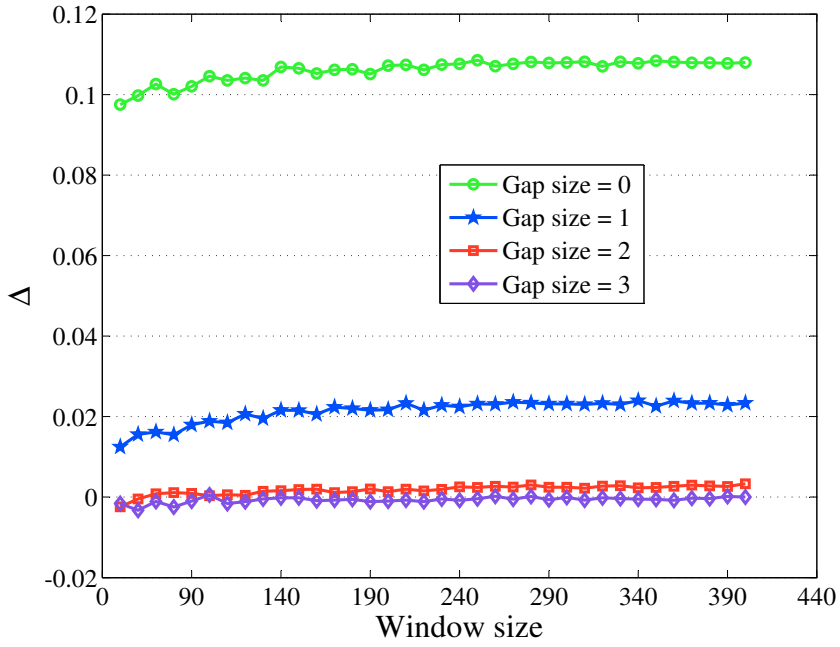


Figure 3.6: Difference of relative occurrence of a particular gap in a CGI and a non-CGI for different window lengths.

Algorithm

Recall that, SONF produces the output, Y_n, by combining maximum signal-to-noise ratio and least squares optimization criteria. The implementation of the the two-fold optimization in SONF approach is shown in Figure 2.9, where the instantaneous matched filter (IMF) is first used to detect the presence of a short duration signal embedded in noise by maximizing the signal-to-noise ratio over variable-time observation interval m. The IMF output, I_n, is then scaled by a locally generated function, Λ_n, using least squares (LS) optimization procedure to obtain the optimal estimate, Y_n, of the message signal S_n. For the implementation of this algorithm, the set of relations (2.17) are utilized choosing the initial value of the gain $P(0)$ to be equal to 1 and assuming $\iota(0) = 0$.

The steps of the proposed SONF based CGI identification scheme for a DNA sequence is given in the following algorithm.

Algorithm 3.1

Initialization: Set the base location index $n = 0$.

- **Step 1:** Apply a rectangular window of length $L = 200$ starting at the base location n of the DNA sequence X, of length N, to obtain the windowed sequence X_n.

- **Step 2:** Obtain the binary indicator sequence X_{CG} for the windowed sequence, X_n, obtained from Step 1.

- **Step 3:** X_{CG} from Step 2, along with the binary basis sequence Φ, given in 3.13, form the inputs to SONF. The corresponding SONF output sequence, Y_n, is evaluated using the set of relations given in (2.17), by assuming $P(0) = 1$ and $\iota(0) = 0$.

- **Step 4:** Compute the SNR gain $G(X_n)$, which is the ratio of the variance of the SONF output Y_n to the variance of the corresponding input X_n.

- **Step 5:** Increment the value of n by 1, i.e., $n = n + 1$. If $n \leq (N - L)$ go to step 1, else go to step 7.

- **Step 6:** Plot $G(X_n)$ as a function of $n + L$ and get its upper envelope. The peaks in the resulting plot which are above a certain choice of threshold, η, indicate the locations of CGIs identified in X.

- **Step 7:** Exit the algorithm.

As an illustration, the various signals involved in the implementation of the proposed SONF scheme is shown in the Figure 3.7. For this purpose, segments of a CGI and non-CGI are considered as shown in the Figure 3.7(a) and Figure 3.7(b) respectively. Naturally, in Figure 3.7(a) there are greater number of ones. Figure 3.7(c) and Figure 3.7(d) show the IMF output for a CGI and a non-CGI respectively. It can be seen that the IMF output corresponding to a CGI progressively increases to a greater value of 35 as compared to 6 of that of a non-CGI. Figure 3.7(e) and Figure 3.7(f) are the scaling functions for a CGI and a non-CGI respectively. They are obtained using the relation $\lambda(m) = P(m)\phi(m)$ in (2.17). Finally, the Figure 3.7(g) and Figure 3.7(h) show the estimated CGI characteristic in a CGI and a non-CGI respectively. The SONF output corresponding to a CGI has greater amplitude as compared with that of a non-CGI.

Note that in the above figures segments of CGI and non-CGI, each of length 80 bp, are shown for the sake of clarity of the illustrations.

3.3 Results and Discussion

The proposed CGI prediction scheme is tested on several genomic sequences of varying lengths taken from the human chromosomes 21 and 22. Specifically, we have used the three contigs, NT_113952.1, NT_113954.1 and NT_113958.2 from chromosome 21, and

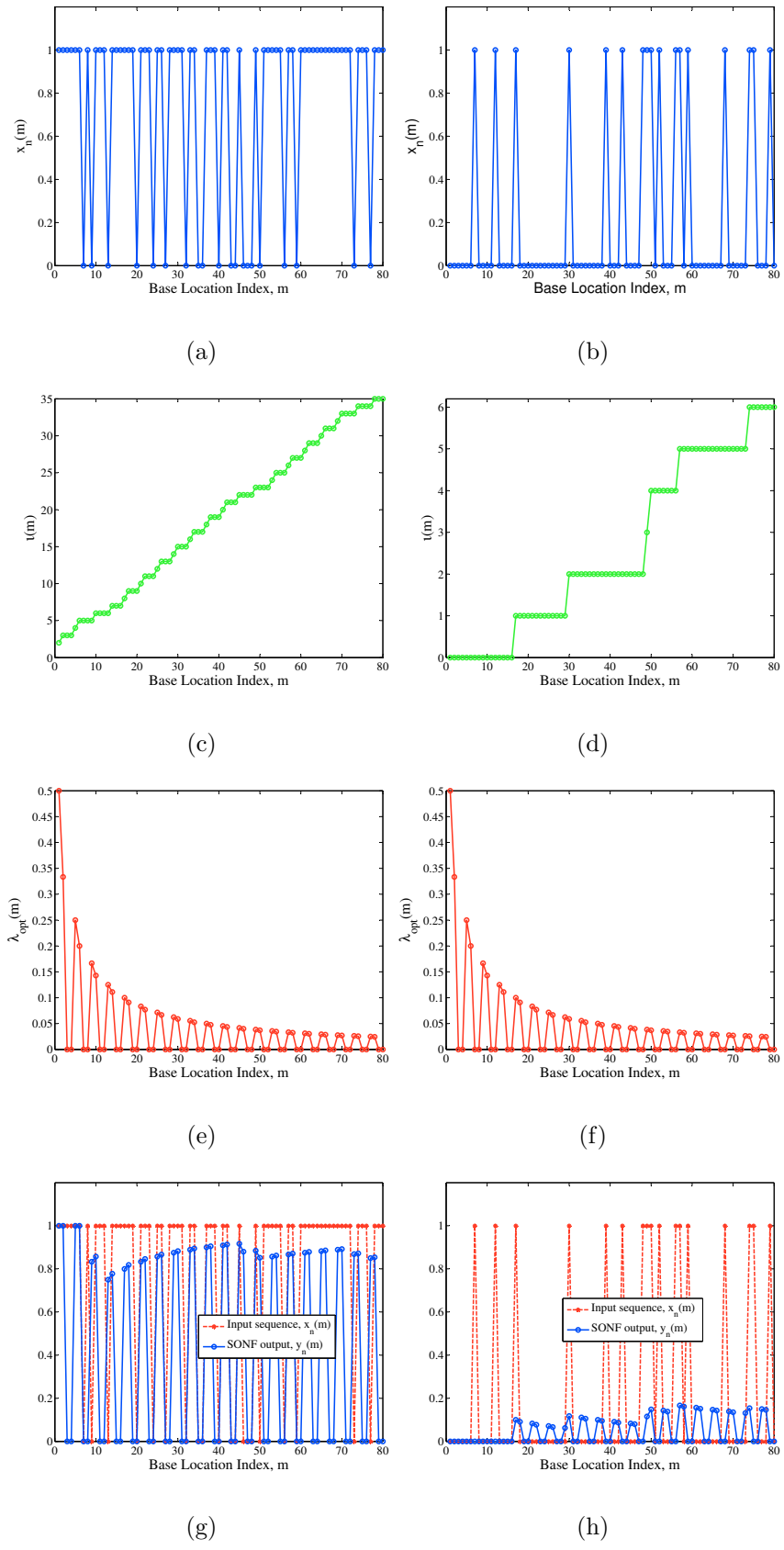


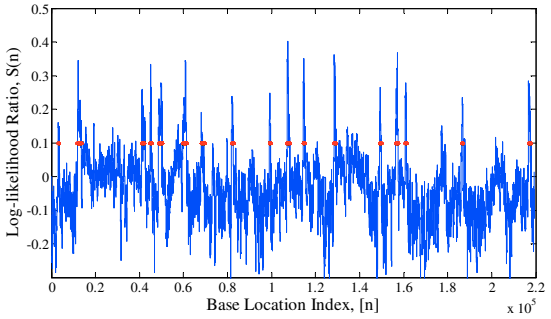
Figure 3.7: SONF implementation. (a) An example of a CGI. (b) An example of a non-CGI. (c) IMF output for CGI. (d) IMF output for non-CGI. (e) Scaling function for CGI. (f) Scaling function for non-CGI. (g) SONF output for CGI, and (h) SONF output for non-CGI.

the contig NT_028395.3 from chromosome 22 for our analysis. All the sequence data considered for this study is obtained from the GenBank database [5]. The performance of the proposed scheme is compared with that of other popular DSP based approaches such as Markov chain [62], IIR low-pass filters [63] and multinomial model [64]. First, a DNA sequence from human chromosome X with the GenBank accession number of L44140 is analyzed for obtaining the values of threshold, η, used by the above methods considered in this study. The sequence is of length 219447 bp, and is already annotated, i.e., the locations of its CGIs are already known and can be obtained from [5].

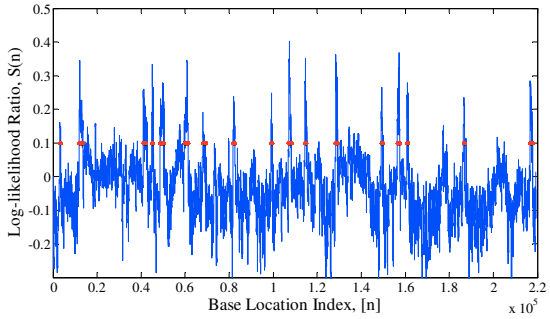
Figure 3.8 shows the comparative performance of CGI prediction by the above mentioned four approaches. Figure 3.8(a) shows the performance of Markov chain approach, where log-likelihood ratio $S(n)$ is plotted against base location index n of the sequence. The transition probability tables given in Table 3.1 and Table 3.2 are used to calculate $S(n)$. All the base locations, n, with $S(n) > 0$ imply that they are very likely to be a part of a CGI. A window length of 200 bp is considered for the method. Markov chain method is able to detect most of the CGIs in the DNA sequence, and it can be seen that the CGIs and non-CGIs can be reasonably differentiated by looking at the sign of $S(n)$. However, one of the major drawbacks of this method is the presence of a lot of false positives that falsely categorize non-CGIs as CGIs.

The Figure 3.8(b) shows the performance of IIR low-pass filter approach where the log-likelihood ratio, $S(n)$, is plotted against the base location index n of the sequence. The transition probability tables given in [63] are used to calculate $S(n)$.

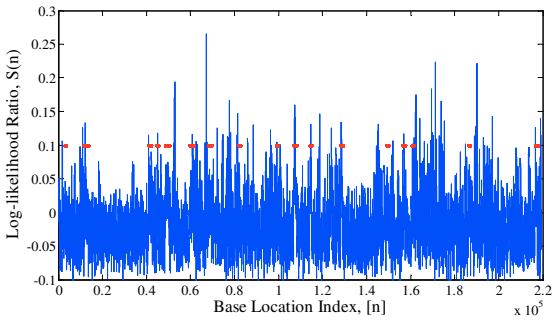
For a fair comparison, instead of a bank on M filters, we have used one pole filter with optimized parameter $\alpha = 0.99$. All the base locations, n, with $S(n) > 0$ imply that they are very likely to be a part of a CGI. A window length of 200 bp is considered for this method. Similar to the Markov chain method, this method also



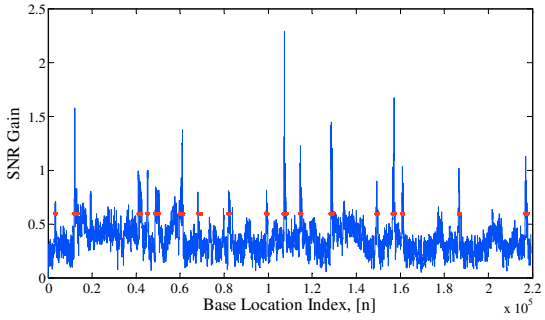
(a)



(b)



(c)



(d)

Figure 3.8: CGI prediction in the DNA sequence L44140 using (a) Markov chain method (b) IIR Filter method (c) Multinomial model (d) SONF scheme.

produces a lot of false positives affecting the prediction accuracy.

Figure 3.8(c) shows the prediction of CGIs using the method of [64], which employs the multinomial model. The multinomial model has been employed in this method to obtain the transition probability tables. A Blackman window of length 100 bp is employed for calculating the filtered log-likelihood ratio. The Blackman window gives larger weights for central samples of the window, thus reducing the edge effects. Windows with the positive filtered log-likelihood ratio are considered to be a part of a CGI. As seen in Figure 3.8(c), this method shows considerably high false positives making the CGI prediction unreliable.

Figure 3.8(d) shows performance of the proposed SONF scheme in predicting the CGIs. Unlike the above mentioned methods, this scheme utilizes the binary basis sequence Φ, given in 3.13, instead of the probability transition tables. Effectiveness of the proposed scheme is clearly seen in Figure 3.8(d), which depict more prominent peaks as compared to the other three approaches. These peaks facilitate more accurate identification of CGIs.

It can be seen from the Figure 3.8, that the default threshold on $\eta = 0$ produces a lot of false positives for the methods using transition probability tables. The optimal threshold values for the methods is obtained by calculating the prediction accuracy (Acc) for varying thresholds for each method (Figure 3.9). The optimal values of thresholds obtained for the Markov chain method, IIR filter method and the proposed SONF approach are 0.1, 0.05 and 0.6 respectively. The true locations of the CGIs, obtained from NCBI website, present in the sequence L44140 are represented by red horizontal spots in Figure 3.8.

The receiver operating characteristic (ROC) curves, shown in Figure 3.10, is obtained for the four methods. It can be seen in Figure 3.10 that the proposed approach has better overall performance for the sequence L44140 with the area under the curve 0.7460. The Markov chain method is next with the area under the ROC curve 0.6072.

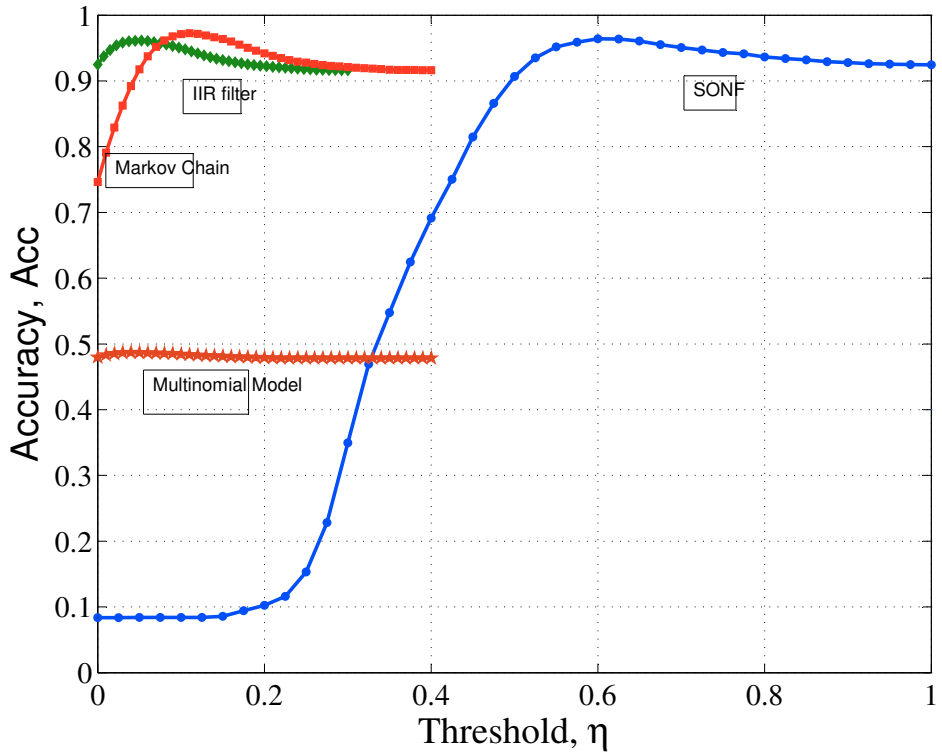


Figure 3.9: Relation between the performance accuracy (Acc) and threshold.

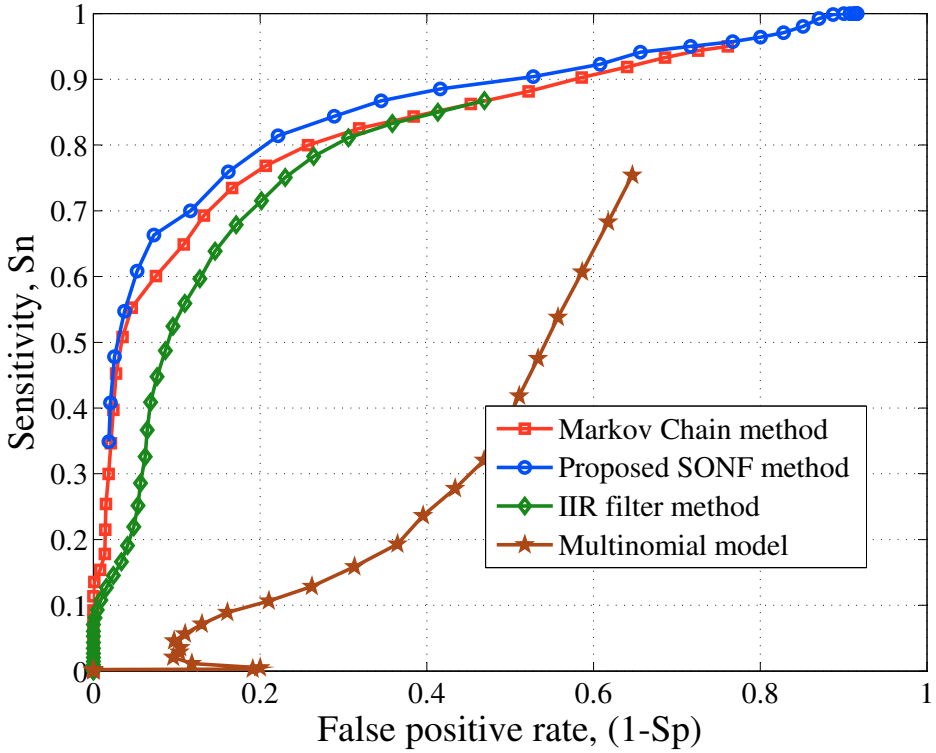


Figure 3.10: ROC curves obtained for the sequence L44140.

The area under the curve for IIR filter method is 0.3106. It can be seen that the multinomial model method has the least area under the ROC curve. The dismal performance of this method is not a reflection on the method itself rather on the use of the transition probability tables.

Figure 3.11, shows only the first 15000 bps of L44140, in Figure 3.8, comparing the prediction of the four methods. The red horizontal lines in Figure 3.11 are the true locations of CGIs. The blue binary decision curve depicts the locations of the predicted CGIs. The binary decision curve for each of the methods is obtained by making the window outputs either equal to 1 or 0 depending upon its value being greater than or less than the corresponding threshold. It can be seen in Figure 3.11(c), that the multinomial based approach fails to detect the CGI located between base pairs 3095 and 3426 as opposed to other three methods implying that the probability transition parameters used for the CGI identification play a crucial role. Hence, it is important to have a CGI identification characteristic which is devoid of any ambiguity considering the choice of different probability transition tables available. The binary basis sequence Φ in the proposed scheme successfully identifies the CGIs and can be reliably used as a CPG identification characteristic.

In this work, the performance of different CGI identification methods is evaluated at the nucleotide level. For example, the value of TP is obtained by adding all the nucleotides predicted to to true positive, and the other outcomes are calculated in the similar manner. Table 4.1 presents the summary of performance measures Sn, Sp, CC and Acc obtained for the analysis of four contigs NT_113952.1, NT_113954.1, NT_113958.2 and NT_028395.3. The performance of the proposed scheme is also compared with that of CpGCluster [18], which uses the distance between CpG dinucleotides (and not the transition probability tables) for identifying CGIs. The proposed approach has the highest values of Sn for all the contigs, and has the high values of CC for the contigs NT_113954.1 and NT_113958.2. The performance accuracy is

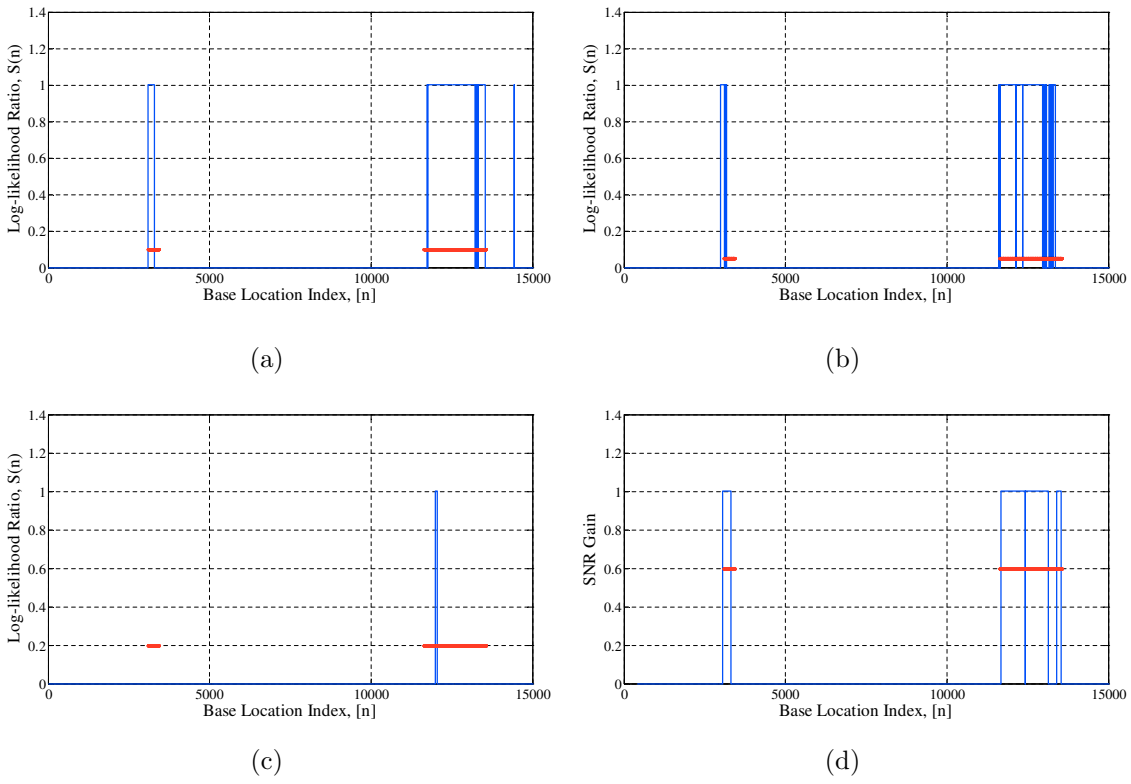


Figure 3.11: CGI prediction in the first 15000 bps of L44140 using (a) Markov chain method (b) IIR Filter method (c) Multinomial model (d) SONF scheme. Binary decision based on respective threshold is plotted against the base location index.

also above 97%.

The above discussion shows that the proposed method is reliable and the proposed binary basis sequence Φ can be used as a CGI identification characteristic. The multinomial method didnt identify any of the CGIs in the contig NT_028395.3 and hence its Sn and Sp values are zero. The corresponding Acc value is high because the method predicting most of the true negatives correctly. The contig NT_028395.3 has short CGIs of the order of 200 bps and the proposed approach with better sensitivity is capable of identifying them.

Table 3.3: Comparison Of Different Methods For Identification Of CGIs

Contig.	**Performace**	**Methods**				
		Markov Chain	**IIR Filter**	**Multinomial model**	**CpGCluster**	**SONF**
NT_113952.1	*Sn*	0.8466	0.8656	0.4524	0.5046	**0.8677**
Length = 184355	*Sp*	0.8728	0.8320	0.2833	0.9995	0.4457
	CC	0.8621	0.8180	0.3609	0.6941	0.6192
	Acc	0.9955	0.9848	0.4948	0.9778	0.9878
NT_113954.1	*Sn*	0.3285	0.2226	0.0055	0.2986	**0.5420**
Length = 129889	*Sp*	0.3082	0.2585	0.0021	0.9946	0.2094
	CC	0.3152	0.2369	0.0040	0.4381	0.4382
	Acc	0.9940	0.9940	0.4989	0.9690	0.9894
NT_113958.2	*Sn*	0.4555	0.3561	0.2938	0.2716	**0.8852**
Length = 209483	*Sp*	0.4652	0.4439	0.0202	0.9994	0.2880
	CC	0.4527	0.3899	0.0119	0.4996	0.4954
	Acc	0.9849	0.9845	0.4960	0.9532	0.9705
NT_028395.3	*Sn*	0.5440	0.4200	0.0000	0.4489	**0.8789**
Length = 647850	*Sp*	0.8233	0.7590	0.0000	0.9947	0.4534
	CC	0.6667	0.5616	-0.0116	0.9753	0.6267
	Acc	0.9945	0.9932	0.8710	0.9532	0.9887

Additionally, we have evaluated the time complexity of the proposed method using the *tic-toc* function in MATLAB. Taking the necessary precautions (such as all applications except MATLAB were closed, a fresh session of MATLAB was started for each run, and MATLAB was warmed up with the code, i.e., the first run of the code was ignored), the CPU time for processing a fixed length of sequence was found to be the least for the Markov chain method. This method was followed by SONF, IIR and multinomial approaches taking an additional CPU time of 1.29%, 1.78% and 1.82% respectively.

3.4 Prediction of Protein Coding Regions

The locations of CGIs help in finding the promoter regions in a DNA sequence. As every gene is preceded by a promoter, finding CGIs in turn help us in determining the locations of genes. The genes in eukaryotic DNA have an alternating arrangement of exons and introns. Predicting the locations of exons in a gene is an important problem as they are responsible for coding proteins. The exons in a gene determine the messenger RNA (mRNA) transcript, which is in turn used for the synthesis of a protein. Sometimes, a single gene can produce multiple proteins due to a process called *alternate splicing*. In this process, particular exons of a gene may or may not be included in the mRNA transcript, as shown in Figure 3.12, resulting in multiple mRNA transcripts. The proteins translated from these alternatively spliced mRNAs will differ in the respective sequence of amino acids. Hence, finding the locations of exons in a gene is an important step in the analysis of DNA sequences as they determine the exact protein they synthesize.

It has been observed that exons in genes exhibit a *period-3* property [93], i.e., the frequency spectrum of DNA segments corresponding to exons tend to exhibit a strong component at the frequency, $2\pi/3$. This property can be attributed to the triplet

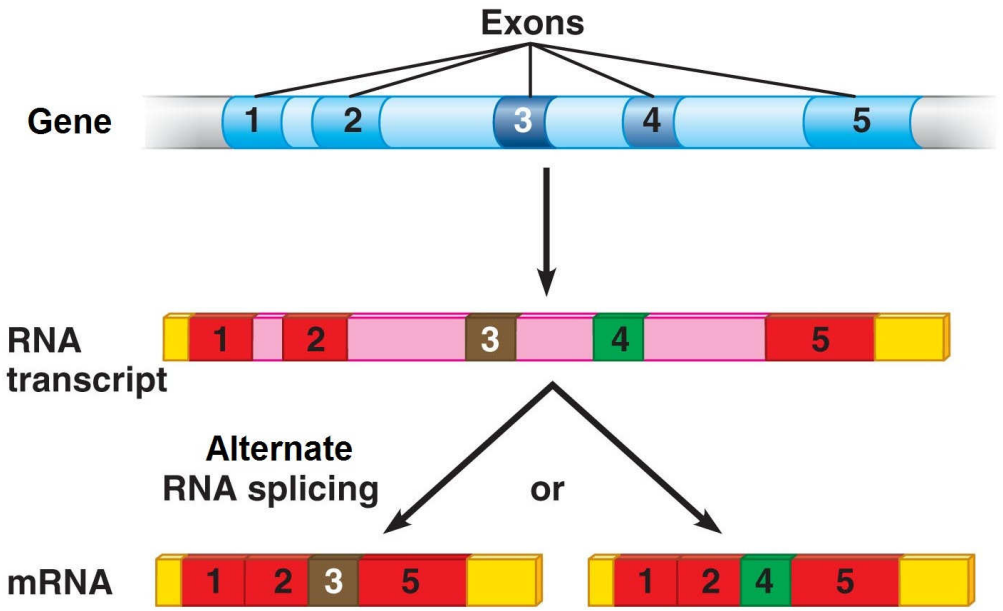


Figure 3.12: Alternative splicing of a gene.

nature of the codons and their unequal usage in coding regions along with the biased usage in genomic DNA [55]. The period-3 property is regarded by researchers as a good preliminary indicator of exon locations, although there are some exceptions in which exons do not satisfy this period-3 property.

3.4.1 Frequency analysis of DNA sequences

A DNA sequence, X, can be mapped into a set of four digital signals X_A, X_T , X_G and X_C using Voss's binary indicator sequences [70] as explained in Section 2.3. Since, exons exhibit the period-3 property, whereas, introns do not, the Fourier spectrum of the binary indicator sequences can be used for predicting the locations of exons [51,53, 55]. For this purpose, a window of size L can be first applied to each binary indicator sequence starting at index $n = 0$. The discrete Fourier transform (DFT) [94] of the windowed sequence of length L can then be computed. The DFT $\mathcal{X}_A(k)$ of the binary sequence, say X_A, is given by

$$\mathcal{X}_{\mathcal{A}}(k) = \sum_{m=0}^{L-1} e^{-j2\pi km/L}, 0 \leq k \leq (L-1), \tag{3.14}$$

where the window length, L, is a multiple of 3. The DFTs of the other binary sequences can be evaluated in a similar fashion. The total magnitude spectrum of the windowed sequence can be obtained as

$$\mathcal{S}(k) = |\mathcal{X}_{\mathcal{A}}(k)|^2 + |\mathcal{X}_{\mathcal{T}}(k)|^2 + |\mathcal{X}_{\mathcal{G}}(k)|^2 + |\mathcal{X}_{\mathcal{C}}(k)|^2, \tag{3.15}$$

where $\mathcal{X}_{\mathcal{T}}(k)$, $\mathcal{X}_{\mathcal{G}}(k)$ and $\mathcal{X}_{\mathcal{C}}(k)$ are the DFTs of the binary indicator sequences X_T , X_G and X_C, respectively.

The period-3 property of exons in a DNA sequence implies that the DFT coefficients corresponding to $k = L/3$ are larger in an exon region as compared to the other $L-1$ coefficients for that window. This process of computing $\mathcal{S}(L/3)$ is repeated by sliding the window by one or more bases at a time. The magnitude spectrum value $\mathcal{S}(L/3)$ can be plotted as a function of the window index n. Peaks occurring in the plot of $\mathcal{S}(L/3)$ versus index n indicate the possible locations of exons as shown in Figure 3.13. Researchers have used other DSP techniques such as the sliding window DFT [55], digital filters [53, 56, 57], wavelet transform [58], and multirate DSP models [59, 60], employing this period-3 property for predicting the locations of exons in DNA sequences.

3.4.2 Proposed SONF based method

In this work, the use of statistically optimally null filters, reviewed in Section 2.4, is proposed to solve the problem of prediction of exons in DNA sequences [90].

Consider an unannotated DNA sequence X, of length N, in which the locations of exons need to be identified. SONFs can be employed to solve this problem by considering the period-3 property of exons as the signal of interest that need to be estimated

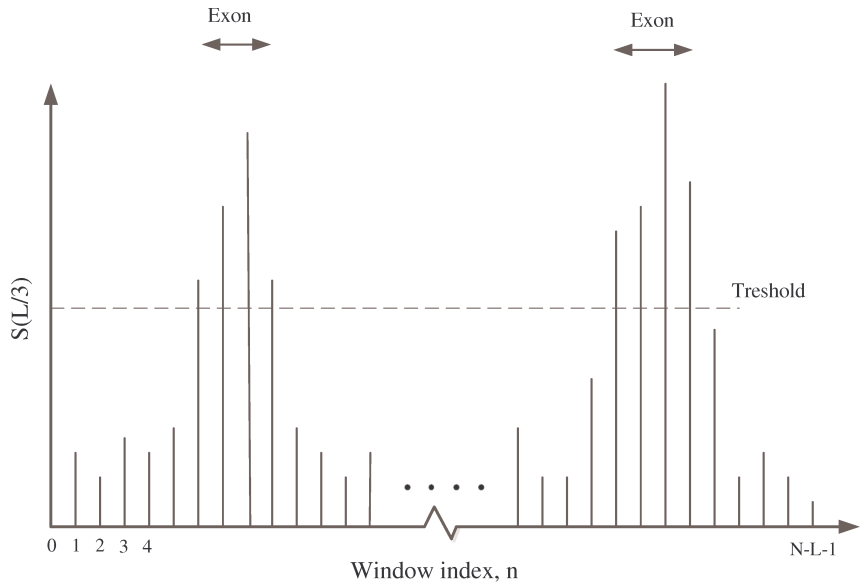


Figure 3.13: Predicting the location of exons using frequency spectrum.

in the DNA sequence X. A sliding window of length L is used to determine whether or not a windowed set of the sequence X, $X_n = \{x_n(m)\}$, where $n = 1, 2, \ldots, N - L + 1$ and $m = n, n+1, \ldots, n+L-1$, belong to an exon. It can be noted that each windowed sequence, X_n, can be expressed as

$$X_n = S_n + R_n, \tag{3.16}$$

where $S_n = \{s(m)\}$ is the period-3 signal, and $R_n = \{r(m)\}$ a residual signal. In order to implement the SONF based approach, the windowed sequence X_n is first mapped into Voss binary indicator sequences X_A, X_T , X_G and X_C. A separate SONF is used to estimate the signal of interest. A basis sequence Φ having some characteristic property of exon is used in the estimation process producing four SONF outputs Y_A, Y_T , Y_G and Y_C as shown in the Figure 3.14. All the four SONF outputs are combined to estimate the period-3 signal in the input windowed DNA sequence. For this purpose, SNR gain, represented as $G(X_n)$, which is the sum of the ratios of variance of each SONF output to the variance of its respective input, is calculated.

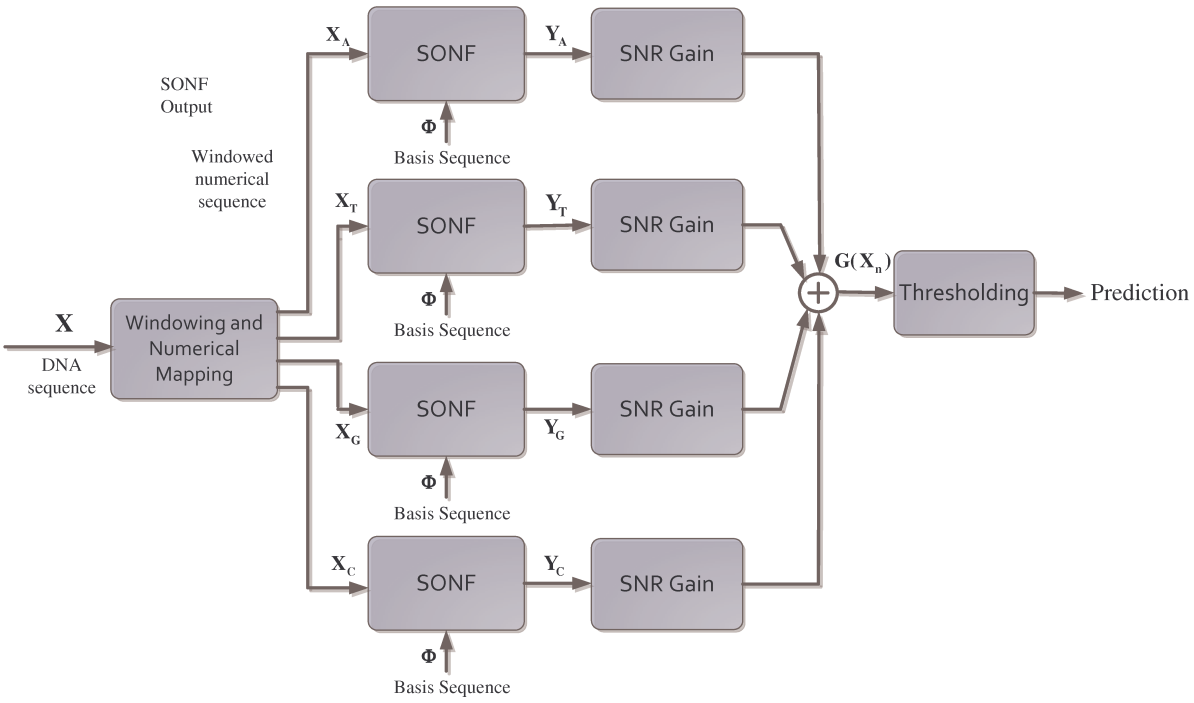


Figure 3.14: A block diagram showing the exon identification algorithm.

Now, by formulating an appropriate threshold on SNR gain, $G(X_n)$, each of the windowed sequences can be classified as belonging to an exon or not.

In the following, we describe a formulation of basis sequence which is appropriate for the detection of the location of exons using SONF.

Formulation of the Basis Sequence

A formulation of the basis sequence, Φ, based on some characteristic properties of exons, is very important for identifying them using SONF approach in the input DNA sequence. As discussed earlier, the protein coding regions (exons) in DNA sequences exhibit period-3 property. This property is believed to be due to a specific periodic arrangement of nucleotides in exons.

The period-3 property of exons in a DNA sequence is reflected in the windowed magnitude spectrum, $\mathcal{S}(L/3)$, through a peak at $k = L/3$. It is seen from (3.15) that this peak in the $\mathcal{S}(k)$ would arise from one or more of its four constituents. In order to

happen this, one or more of the binary sequences themselves should therefore have the period-3 property. Thus, the basis function must be chosen to have the capability of capturing this feature in a binary sequence, if this sequence indeed has such a feature. In view of these reasons, a sequence of length L given by

$$\phi = \{100100100 \ldots 100100\}, \tag{3.17}$$

could be a reasonable choice for the basis function. In this sequence '1' can then be expected to capture the presence and '0' the absence of the nucleotide that gives rise to particular binary sequence. It is noted that this choice of the basis function satisfies the period-3 property and its magnitude spectrum is marked by a peak at $k = L/3$, as seen from Figure 3.15. However, a DNA sequence has three reading frames, and in an windowed DNA sequence belonging to an exon, the period-3 property could arise from any of these three reading frames. Due to this reason, the basis function $\Phi = \{\phi_1, \phi_1, \phi_1\}$ containing an orthogonal set of sequences, each having the period-3 property, given by

$$\begin{aligned} \phi_1 &= \{100100100 \ldots 100100\} \\ \phi_2 &= \{010010010 \ldots 010010\} \\ \phi_3 &= \{001001001 \ldots 001001\} \end{aligned}$$

is chosen to predict the protein coding regions using SONF.

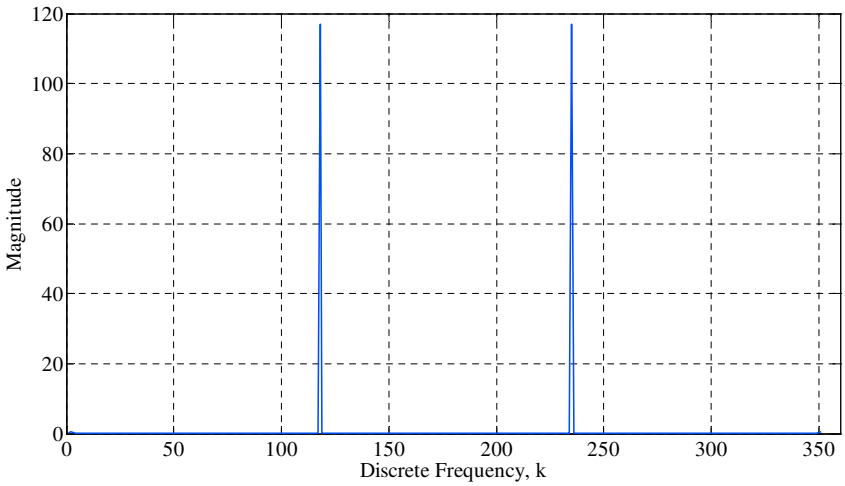


Figure 3.15: Frequency spectrum of the basis sequence Φ.

Algorithm

The steps of the proposed SONF based exon identification scheme for a DNA sequence is given in the following algorithm.

Algorithm 3.2

Initialization: Set the base location index $n = 0$.

- **Step 1:** Apply a rectangular window of length $L = 300$ starting at the base location n of the DNA sequence X, of length N, to obtain the windowed sequence X_n.

- **Step 2:** Obtain the four Voss's binary indicator sequences X_A, X_T , X_G and X_C for the windowed sequence, X_n, obtained from Step 1.

- **Step 3:** Each of the four binary signals from Step 2, along with the binary basis sequence Φ, given in 3.17, form the inputs to an SONF resulting in four SONF output sequence Y_A, Y_T , Y_G and Y_C. These are evaluated using the set of relations given in (2.17), by assuming initial P to be an identity matrix of order 3 and $\iota(0) = 0$.

- **Step 4:** Compute the sum of the SNR gains for each of the SONF outputs from Step 3 to obtain $G(X_n)$.

- **Step 5:** Increment the value of n by 1, i.e., $n = n + 1$. If $n \leq (N - L)$ go to step 1, else go to step 7.

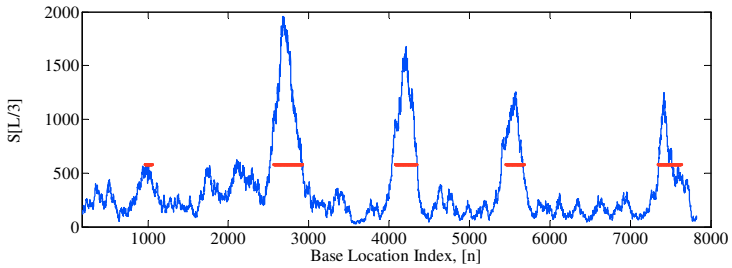
- **Step 6:** Plot $G(X_n)$ as a function of $n + L$ and get its upper envelope. The peaks in the resulting plot which are above a certain choice of threshold, η, indicate the locations of exons identified in X.

- **Step 7:** Exit the algorithm.

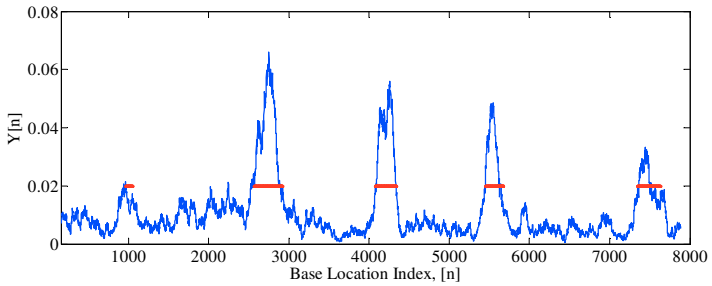
3.5 Results and Discussion

The proposed exon prediction scheme is tested on the DNA sequences taken from the chromosome III of *C. elegans*. The performance of the proposed scheme is compared with that of other popular DSP based approaches such as DFT method [55] and anti notch filters [95]. We have used the DNA sequence containing the gene with geneID F56F11.4 taken from GenBank [5] for our analysis. This sequence is analyzed for obtaining the values of threshold, η, used by the above methods considered in this study. The sequence is of length 8000 bp, and has five protein coding regions whose locations are reported in [5].

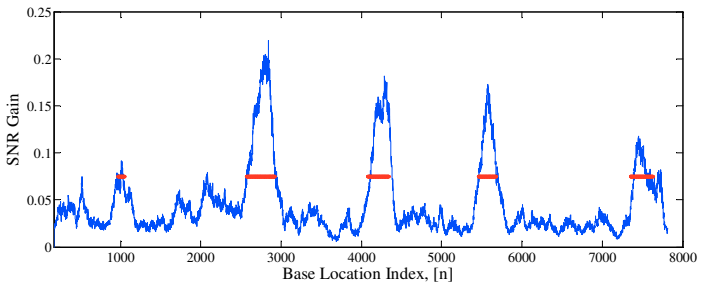
Figure 3.16 shows the comparative performance of exon prediction by the above mentioned three approaches. Figure 3.16(a) shows the performance of DFT based method, where $S(L/3)$ is plotted against base location index n of the sequence. A window length of 351 bp is considered to be appropriate for predicting exons [51]. The peaks in the spectrum correspond to regions where three base periodicity is dominant. The Figure 3.16(b) shows the performance of anti notch filter approach where the output, $Y(n)$, is plotted against the base location index n of the sequence.



(a)



(b)



(c)

Figure 3.16: Exon prediction in the gene F56F11.4 using (a) DFT method. (b) Anti notch method. (c) Proposed SONF scheme.

Finally, Figure 3.16(c) shows performance of the proposed SONF scheme in predicting the exons. The red horizontal lines in Figure 3.16 are the actual locations of the exons, and are plotted at height equal to the threshold values of 580, 0.02 and 0.075 used for the respective method. The first coding region in F56F11.4 is very short and is of length 112 bp. It can be seen from Figure 3.16 that the proposed SONF based method is capable of predicting this exon more accurately. The performance metrics of the prediction results obtained for the sequence F56F11.4 are given in Table 3.4. The high values of sensitivity and the correlation coefficients show the effectiveness

Table 3.4: Comparison of Different Exon Prediction Methods

Methods	**Performance Criteria**		
	Sensitivity S_n	**Specificity** S_p	**Correlation Coefficient** CC
DFT method	0.8676	0.8375	0.8483
Anti notch filter	0.8202	0.8036	0.8122
Proposed SONF scheme	0.8994	0.8622	0.8694

of the proposed SONF based method over the other two methods.

An exhaustive analysis is done on the DNA sequences taken from the chromosome III of *C. elegans* using the three methods. The receiver operating characteristic (ROC) curves, shown in Figure 3.17, is obtained as a result of this analysis. It can be seen in Figure 3.17 that the proposed approach has better overall performance for the sequence F56F11.4 with the area under the curve 0.8206. The DFT method is next with the area under the ROC curve 0.7872. The area under the curve for anti notch filter method is 0.7519.

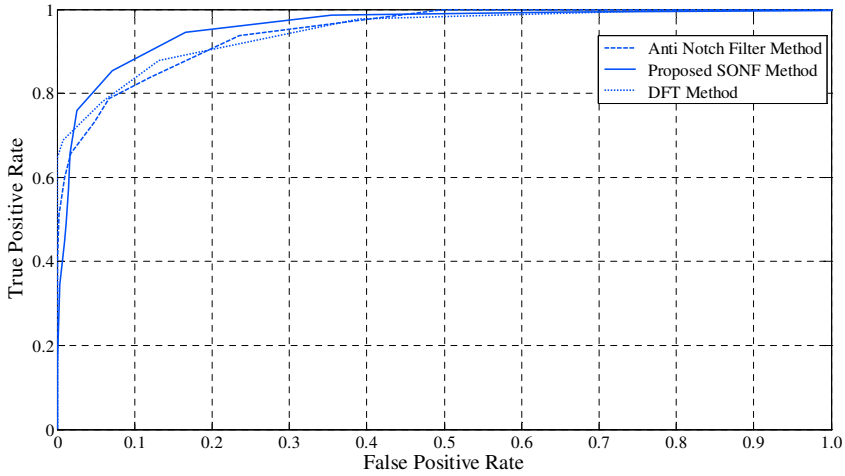


Figure 3.17: The ROC curves of the exon prediction methods.

3.6 Summary

In this chapter, DNA sequences have been analyzed in order to investigate the problems of identifying the locations of CpG islands and protein coding regions (exons).

For the problem of identifying the locations of CGIs, an SONF based approach has been proposed. For this purpose, a basis function has been formulated having a characteristic property of a CGI, i.e., the G and C content is $\geq 50\%$ and the nucleotide G tends to immediately follow C in a CGI. This basis sequence has then been used in SONF, to identify the locations of the CGIs. SONF is implemented using two-fold optimization of maximizing the signal-to-noise ratio and least square optimization. The instantaneous matched filter, which maximizes the signal-to-noise ratio, is first used to detect the presence of CGI followed by application of least squared optimization to obtain the optimal estimate of the signal pertaining to CGIs. It has been shown that unlike the use of the transition tables that are dependent on training data, the proposed basis sequence is more reliable in identifying CGIs. The performance of the proposed technique for the prediction of CGIs has been tested on four randomly chosen contigs in chromosomes 21 and 22 of human beings. The results obtained have been shown to be more accurate than those obtained using the existing methods.

The problem of predicting protein coding regions (exons) in DNA sequences has also been investigated using SONF. For this purpose, a basis function having a characteristic property of an exon has been formulated. The period-3 property exhibited by exons has been chosen to formulate the basis sequence. The DNA sequence is mapped to the four binary indicator sequences each of which is processed by a separate SONF to capture its period-3 property individually. The performance of the method developed has been compared with the other existing methods for predicting exons in DNA sequences. It has been shown that the proposed algorithm is quite effective especially in predicting short exons.

Chapter 4

Analysis of RNA Sequences

4.1 Introduction

Ribonucleic acid (RNA) is a working copy of DNA resulting form a process known as transcription based on the information contained in DNA. The main task of RNA is to transfer the genetic information contained in DNA from nucleus to ribosome for the creation of proteins. There are different types of RNAs that play various kinds of roles in synthesizing proteins. For example, messenger RNA (mRNA) regulates how the genes in DNA sequences are expressed, transfer RNA (tRNA) carries amino acids in the cell during the process of translation, and ribosomal RNA (rRNA) helps in putting amino acids together in chains forming protein sequences. Apart from the above cellular roles, RNAs also have structural and catalytic roles. Another type of RNA, called the non-coding RNAs (ncRNAs) [96–98] do not code for proteins but play a vital role in various biological functions such as chromosome replication, RNA modification, etc. Due to the above reasons, the study of RNAs has become pivotal to understand fully the biological processes of complex organisms.

A molecule of RNA consists of a sequence of nucleotides attached to one another

by covalent chemical bonds. The nucleotides contain one of the four bases: adenine (A), cytosine (C), guanine (G) or uracil (U). RNA is very similar to DNA except that in RNA the nucleotide uracil (U) replaces thymine (T) in DNA, and RNA is normally found as a single-stranded molecule, whereas DNA is double stranded. The linear sequence of nucleotides in an RNA molecule is called its *primary structure*. An RNA sequence has a property of folding and twining about itself such that the nucleotides in close proximity form weak chemical bonds (hydrogen bonds) with another if they are complementary. The set of nucleotide-pairs existing in a RNA is called its *secondary structure*. An example of an RNA secondary structure resulting in stem and loop patterns is shown in the Figure 4.1. The complimentary nucleotide base pairs (Watson-Crick pairs) are the base A bonding with U and similarly, G with C. The folding and twining of RNA sequence about itself imparts it a stable three-dimensional structure, called the *tertiary structure*. Similar to proteins, there is a correlation between RNA structure and its functionality [62]. Predicting the secondary structure of an RNA sequence is an initial step in predicting its tertiary structure and consequently its functionality. Two of the main substructures present in an RNA are loop and stem patterns as seen from Figure 4.1. Another important substructure of an RNA secondary structure, called the pseudoknot, is very common in all classes of RNAs. Pseudoknots are involved in several biological processes [26] and play a crucial role in the determination of RNA tertiary structure. The problem of predicting pseudoknots in RNA secondary structure is very crucial and its complexity is considered to be NP-hard [39].

In this chapter, an efficient and reliable technique for predicting the secondary structure of RNA sequences including pseudoknots using a matched filtering approach is presented [99, 100].

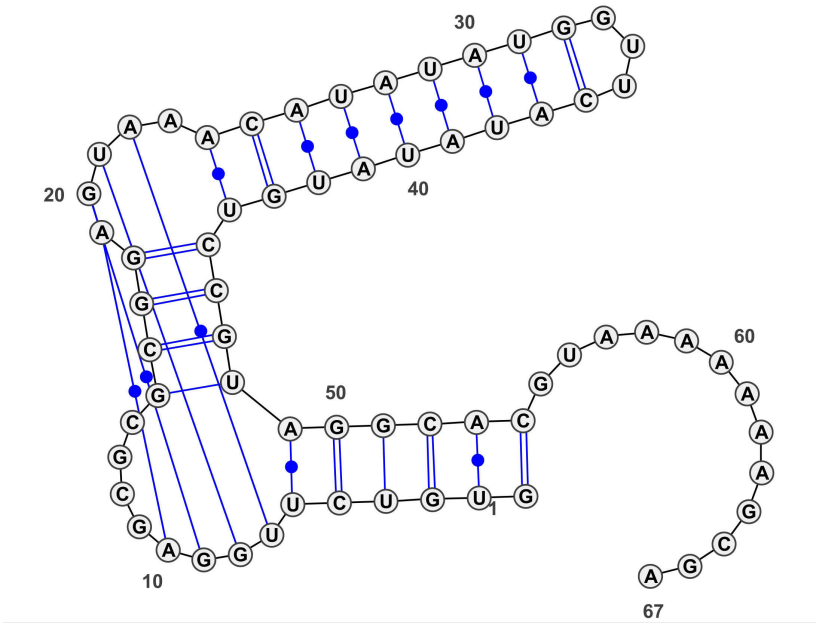
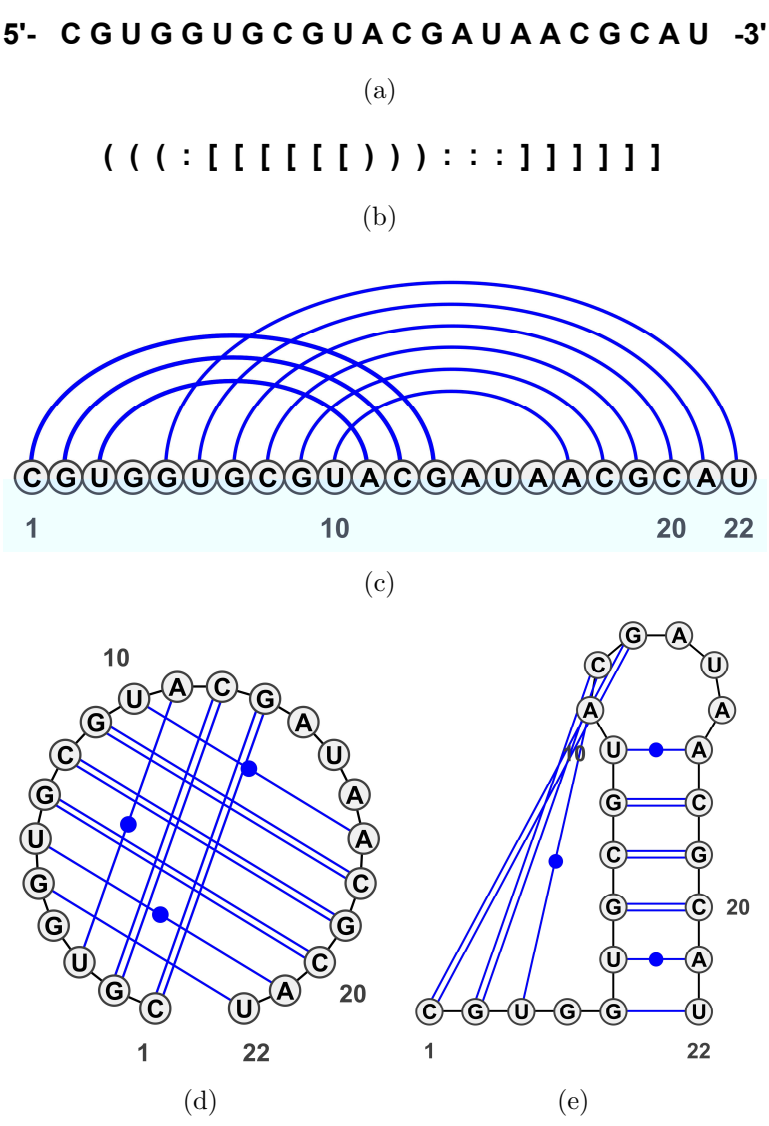


Figure 4.1: RNA secondary structure of the sequence Tomato_mosaic_virus.1.

4.2 RNA Secondary Structure

As mentioned in Section 4.1, an RNA sequence has a tertiary structure and its study is greatly simplified by just concentrating on its secondary structure, i.e., the nucleotide base pairs involved. Consider an RNA sequence of length N, $X = x(1), x(2), \cdots, x(N)$, where $x(i) \in \{A, C, G, U\}\ \forall\ i = 1, 2, \cdots, N$. For $1 \leq i < j \leq N$, let $x(i) \cdot x(j)$ denote the pairing of base $x(i)$ with $x(j)$. The secondary structure S, of the sequence X is a set of base pairs P such that a base is paired with at most one other base. Alternatively, any two base pairs in S, $x(i) \cdot x(j)$ and $x(i^{'}) \cdot x(j^{'})$ are either identical, or else $i \neq i^{'}$ and $j \neq j^{'}$. There can be several valid secondary structures for a RNA primary structure. However, most of the possibilities can be easily eliminated using chemical and stereochemical constraints. Most of these constraints may be formulated in terms of the thermodynamic instability of structures containing certain base-pairs or sets of base-pairs. Such constrains can be utilized in developing algorithms which maximize the stability of RNA structure.



(a)

(b)

(c)

(d)

(e)

Figure 4.2: RNA having a pseudoknot. (a) Primary structure. (b) Bracket notation. (c) Linear representation. (d) Circular representation. (e) Radiate representation.

4.2.1 Substructures of RNA secondary structure

An RNA secondary structure S is composed of substructures or components such as loops, stems, bulge, pseudoknots, interior loops, exterior loops, etc. This work considers only on three substructures: loops, stems and pseudoknots shown in Figure 4.2. A *loop* is an unpaired section of an RNA sequence that is created when it folds and forms base pairs with another section of the same sequence. A *stem* is formed by several stacked base pairs such as $x(i) \cdot x(j)$, $x(i+1) \cdot x(j-1)$, $\cdots$, $x(i+m) \cdot x(j-m)$, where $m > 0$ and $(j - i - 2m) > 0$. And finally, a necessary and sufficient condition for an RNA secondary structure to contain a *pseudoknot* is to have two base pairs $x(i) \cdot x(j)$ and $x(i^{'}) \cdot x(j^{'})$ in the structure such that $i < i^{'} < j < j^{'}$ or $i^{'} < i < j^{'} < j$. This condition causes crossing of base-pairs resulting in a twisted/knotted structure.

4.2.2 Representation of RNA secondary structure

There are a number of ways of representing an RNA secondary structure. The RNA primary structure (Figure 4.2(a)) can be represented using bracket notation as shown in Figure 4.2(b). In this bracket notation, the base-pairs are represented by the corresponding opening and closing brackets, and the unpaired bases are represented by colon. The stack of successive opening or closing brackets corresponds to a stem pattern and the colons correspond to a loop pattern. In the linear representation (Figure 4.2(c)) the RNA molecule is stretched into a line and circular arcs are used to represent the base-pairs. The presence of a pseudoknot is suggested by the intersection/crossing of the arcs joining the base-pairs. In circular representation the bases of the RNA molecule are placed equidistant to one another along the circumference of a circle and the base-pairs are represented by chords as shown in Figure 4.2(d). The presence of a pseudoknot is suggested by the intersection of the cords joining the base-pairs. Figure 4.2(e) shows the radiate representation in which the difference

between a stem pattern and a pseudoknot is better visualized.

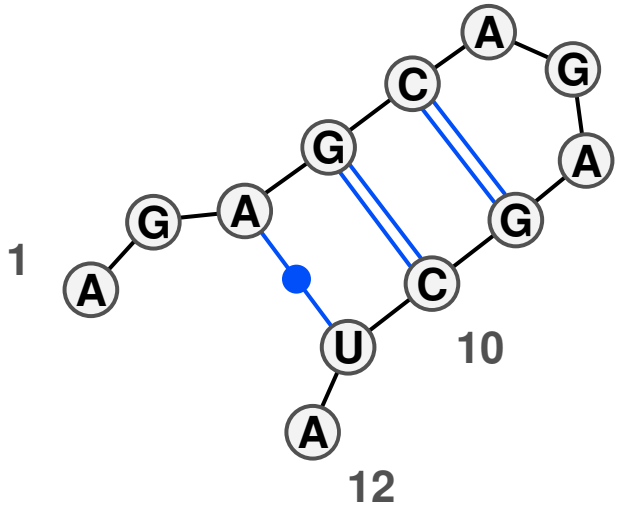
4.3 Proposed Technique

It is well established that an RNA sequence X assumes the most energetically stable configuration. In other words, X assumes a configuration with minimum free energy - which is the energy stored in the chemical bonds of a molecule. As the RNA folds, some bases form bonds with others forming stems and some remain free forming loops. The stem patterns tend to stabilize the RNA structure, where as, the loops tend to destabilize it. Of the possible several sets of secondary structrues of X, the challenge is to predict the most stable structure. Stability of an RNA molecule can be assessed by calculating its free energy. Stems have negative free energy and loops have positive free energy [29]. Consequently, searching for long possible stem patterns in X can lead us to the most energetically stable structure. Predicting the secondary structure of an RNA sequence involves predicting the number of stems, the number of loops, and the presence of pseudoknots, if any, given an RNA primary structure.

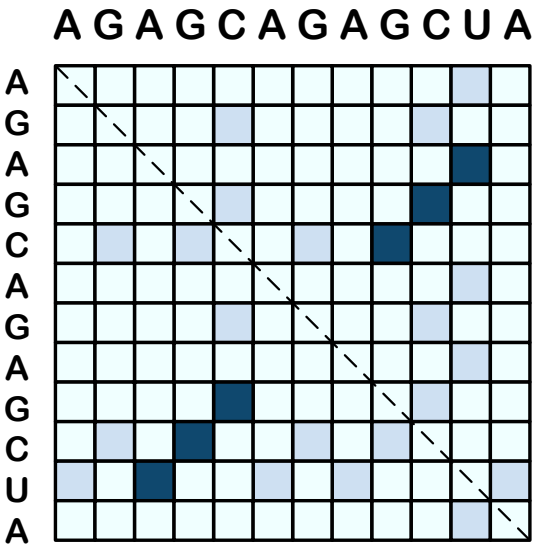
In the proposed approach, we utilize the base-pair matrix representation of an RNA sequence of length N, $X = x_1, x_2, \cdots, x_N$ where $x_i \in \{A, G, C, U\}$. The base-pair matrix B of the above RNA sequence X is an $N \times N$ matrix, and is formulated such that its $(m, n)^{th}$ element b_{mn} has a value of either 0 or 1 according to the criteria

$$b_{mn} = \begin{cases} 1 & \text{if } x_m \text{ and } x_n \text{ form base pair} \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

An example of a base-pair matrix B is shown in the Figure 4.3. All the locations having a value of 1 are shaded in the matrix. The remaining locations have a value of 0. It can be noticed that the matrix is always symmetric and the number of diagonal patterns of shaded locations in either the upper triangle or lower triangle



(a)



(b)

Figure 4.3: Diagonal stem patterns in RNA secondary structure. (a) Radiate representation. (b) Base-pairing matrix. Note: The non-zero elements corresponding to the base-pairs are shaded.

of B determines the possible number of stem patterns in the secondary structure of RNA shown in Figure 4.3(a). In the Figure 4.3(b), the stem pattern is shaded in dark color. Gaps between the diagonal stem patterns corresponding to the upper and lower triangular matrix of B determine the length of loops. A stem pattern can never intersect the dotted diagonal line drawn in the Figure 4.3(b). This constraint can used to validate if the diagonal elements identified in base pairing matrix are really stem patterns or not. Now, in order to identify the possible stem patterns in an RNA sequence, an efficient approach is needed to get the locations of the diagonal patterns in the matrix B. A two dimensional convolution of the upper triangular matrix B_u, obtained from B, with a diagonal matrix D of size $M \times M$ given by

$$C = B_u * D \tag{4.2}$$

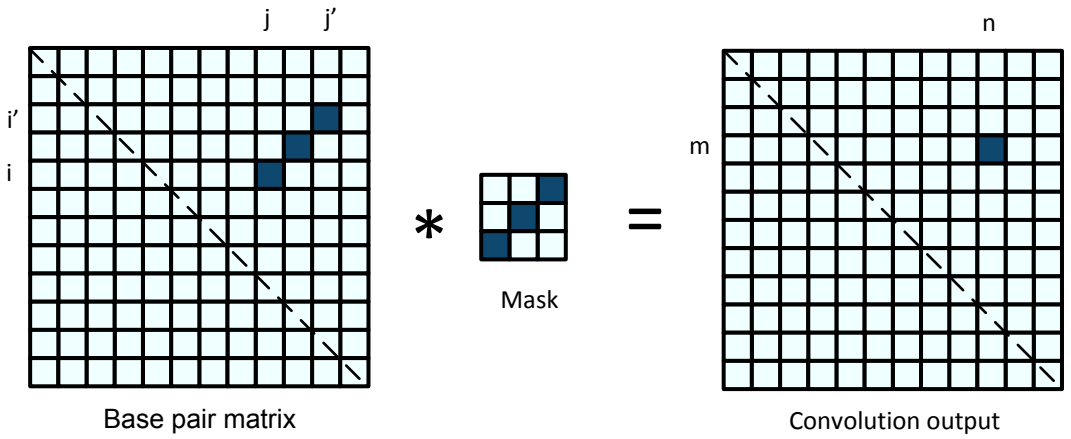
can be used to locate the stem patterns in B. Mathematically, the discrete 2D convolution operation is defined as

$$C(m,n) = \sum_{j=1}^{M} \sum_{i=1}^{M} B_u(m+i, n+j) \cdot D(i,j). \tag{4.3}$$

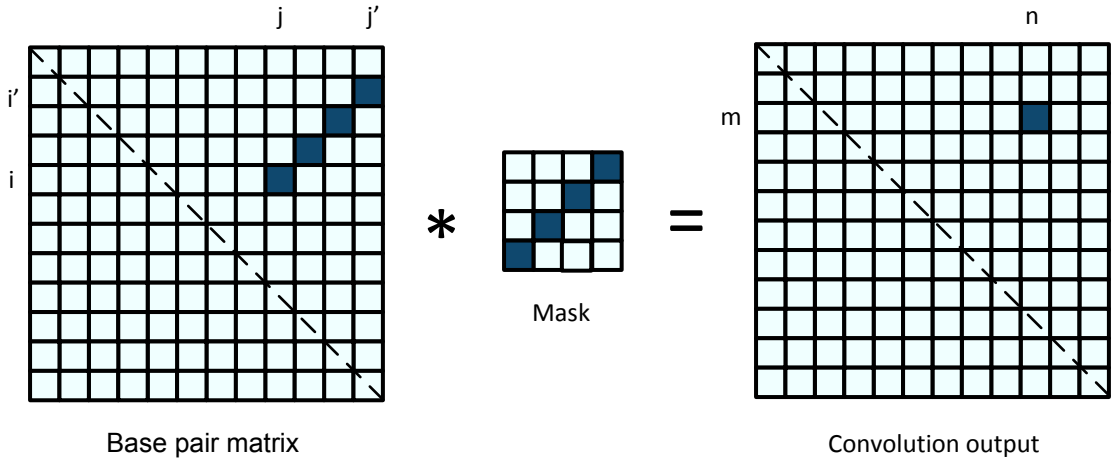
This operation is shown in the Figure 4.4 for both the cases when M is odd and even respectively. The location of the maximum value in the convolution output C is shaded in Figure 4.4(a) and in Figure 4.4(b). If the size of the mask, M, is equal to the length of the stem pattern in an RNA secondary structure, then the maximum value of any element in C is M. Hence, by varying the size of mask, M, the stem patterns of different sizes present in an RNA can be identified.

4.3.1 Prediction of stem and loop patterns

By varying the size of M, i.e., the size of matrix D (also called as mask), the locations of stems of various length can be determined. For the mask of size odd, and if the



(a)



(b)

Figure 4.4: Matched filtering. (a) Using a mask of odd size. (b) Using a mask of even size.

location of the maximum element of C is (m, n) then, the terminal locations of the stem pattern, (i, j) and $(i^{'}, n^{'})$, are given by,

$$(i, j) \rightarrow (m + int(M/2), n - int(M/2)) \qquad \text{and} \tag{4.4}$$

$$(i^{'}, n^{'}) \rightarrow (m - int(M/2), n + int(M/2)) \tag{4.5}$$

respectively. Here, $int(M/2)$ rounds the value of $M/2$ to the nearest integer less than or equal to $M/2$.

For the mask of size even, and if the location of the maximum element of C is (m, n) then, the terminal locations of the stem pattern, (i, j) and $(i^{'}, n^{'})$, are given by

$$(i, j) \rightarrow (m + (M/2), n - (M/2) + 1) \qquad \text{and} \tag{4.6}$$

$$(i^{'}, n^{'}) \rightarrow (n + (M/2), m - (M/2) + 1) \tag{4.7}$$

respectively.

The terminal locations of a stem pattern are sufficient to determine the locations of the remaining bases in the stem. Now, all the base-pairs (i, j) corresponding to the stem pattern identified are given by

$$(m - int(M/2) \leq i \leq (m + int(M/2) \qquad \text{and} \tag{4.8}$$

$$(n - int(M/2)) \leq j \leq (n + int(M/2)) \tag{4.9}$$

for M being odd. From the values of i's and j's obtained from the equations (4.8) and (4.8), the base-pairs (i, j) are generated by associating the lowest value of i with

the highest value of j, continuing the process until the highest value of i is associated with the lowest value of j.

And, all the base-pairs (i, j) corresponding to the stem pattern, when M is even, are given by

$$(m + M/2) \leq i \leq (n + (M/2)) \qquad \text{and} \tag{4.10}$$

$$(n - (M/2) + 1) \leq j \leq (m - M/2 + 1)). \tag{4.11}$$

From the values of i's and j's obtained, the base-pairs (i, j) are generated in the similar fashion to the case of odd M. After all the base-pairs are calculated, the locations of the nucleotides in a loop, l, are obtained using $max(i) < l < min(j)$.

For example, if $M = 3$, and the location of $max(C) = 3$ is at $(4, 10)$, then the terminal locations of the stem pattern are $(5, 9)$ and $(3, 11)$. Now, the locations of all the base-pairs forming stem of size 3 are given as $x_3 \cdot x_{11}$, $x_4 \cdot x_{10}$ and $x_5 \cdot x_9$. Similarly, when $M = 4$, and if the location of $max(C) = 4$ is at $(3, 10)$, then the terminal locations of the stem pattern are $(5, 9)$ and $(2, 12)$. Now, the locations of all the base-pairs forming stem of size 4 are given as $x_2 \cdot x_{12}$, $x_3 \cdot x_{11}$, $x_4 \cdot x_{10}$, and $x_5 \cdot x_9$. The nucleotide bases x_6, x_7 and x_8 form the loop.

4.3.2 Prediction of pseudoknots

The 2D convolution is repeatedly calculated by reducing the size of D each time starting with the size $int(N/2)$ until $M \geq 3$. Each time a stem pattern is identified, the base pairing matrix B is modified eliminating the columns corresponding to the stem pattern found. Then the convolution is carried on the remaining columns of B with reduced mask size M. The locations of the stem patterns found are used to update the RNA structure (bracket notation) after each iteration. When $M = 2$,

the algorithm exits and outputs the RNA secondary structure in the form of bracket notation.

From the set of all base pairs in the RNA secondary structure predicted from the stem patterns obtained using convolution, the presence of a *pseudoknot* is determined if there exist two base pairs $x(i) \cdot x(j)$ and $x(i^{'}) \cdot x(j^{'})$ in the structure such that $i < i^{'} < j < j^{'}$.

The following is the summary of the proposed algorithm.

Algorithm 4.1

Initialization: Input the RNA sequence X of length N. Obtain the mask D of size $M = int(N/2)$.

- **Step 1:** Obtain the base pair matrix B and the upper triangular matrix B_u of the input RNA sequence, X.
- **Step 2:** Evaluate the 2D convolution $C = B_u * D$ and find the location, (m, n), of the largest element in C.
- **Step 3:** Using the values of m and n obtained in Step 2, and M being odd or even, find the terminal locations,(i, j) and $(i^{'}, n^{'})$, of the probable stem pattern.
- **Step 4:** Validate the obtained stem pattern in Step 3 by checking if it intersects the diagonal of B_u.
- **Step 5:** If the stem pattern in Step 3 passes the validation, find all the base pairs in the stem pattern for M being odd or even.
- **Step 6:** Modify the matrix B_u eliminating the columns corresponding to the stem pattern found in Step 4.
- **Step 7:** Reduce the size of the mask to be equal to the largest element in C, i.e., $M = max(C)$ and go to Step 2.

- **Step 8:** Continue the procedure until $M \geq 3$, recording the base pairs obtained in the form of bracket notation.
- **Step 9:** The presence of the pseudoknot is detected by checking if there exist two base pairs $x(i)\cdot x(j)$ and $x(i^{'})\cdot x(j^{'})$ in the structure such that $i < i^{'} < j < j^{'}$.
- **Step 10:** Exit the algorithm for $M = 2$.

4.4 Results and Discussion

The performance of the proposed algorithm is validated by testing it against several different RNAs, specially containing pseudoknots. For this purpose, we have used the RNA sequences from PseudoBase database [101]. PseudoBase is a database containing structural, functional and sequence data [102] related to RNA pseudoknots. For each pseudoknot in the database, information such as the number of stems and the number of loops in the pseudoknot is reported along with other information such as the EMBL accession number of the sequence. Results obtained from the proposed approach are compared with the actual secondary structure information given in the PseudoBase database.

The screen shot shown in the Figure 4.5 is the MATLAB based tool developed implementing the proposed RNA secondary structure prediction algorithm. The screen shot shows the prediction result obtained for the RNA sequence PKB111 taken from [101]. Figure 4.6 shows the base pair matrix for the sequence PKB111. In Figure 4.5, the field 'Input RNA Sequence' is used to input the primary structure of the RNA sequence whose secondary structure is to be determined. The 'Process' button below the input field is to start the prediction process. The prediction output is generated in the form of bracket notation and is displayed in the field 'Bracket Notation'.

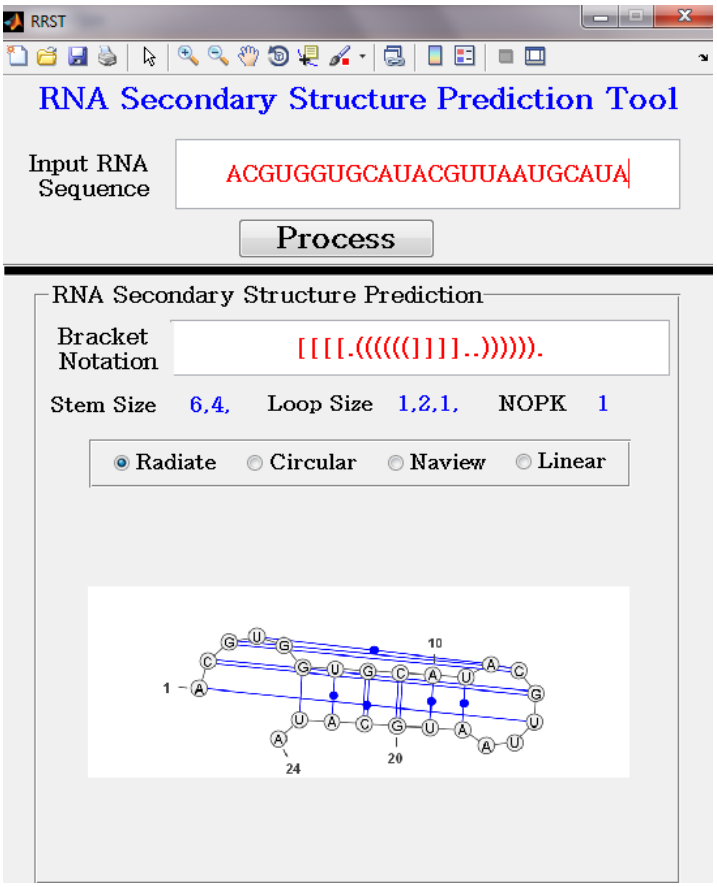


Figure 4.5: A screen shot of the RNA secondary structure prediction tool developed showing the input RNA sequence, the secondary structure output in both bracket notation and radiate notation.

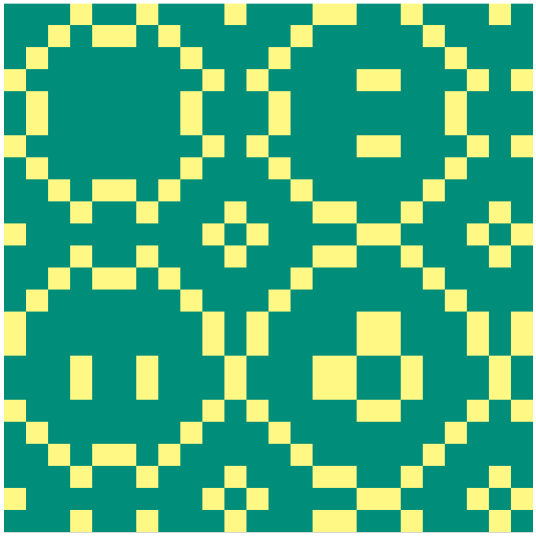


Figure 4.6: Base pairing matrix representation of the sequence PKB111.

At the bottom of the tool there is a provision to visualize the predicted RNA secondary structure in different representations such as Radiate, Circular, Naview and Linear. The VARNA [103] java applet is used to create the visualizations in MATLAB. The tool also displays the number of stem patterns, number of loops, and the number of pseudoknots (NOPK in the tool) in the input RNA sequence. The tool developed is tested on several RNA sequences taken from PseudoBase [101]. The test sequences were chosen such that their length doesnot exceed 100 bases and contain atleast one pseudoknot. The prediction results obtained are validated with the standard PseudoBase results. Both the prediction results and the PseudoBase results are in bracket notation.

In order to quantify the results, the prediction accuracy in terms of sensitivity, Sn, and specificity, Sp and correlation coefficient CC were calculated. Table 4.1 enumerates the prediction results of the proposed algorithm compared with that of the recent popular methods such as HotKnots [104], DotKnot [105], RNAalifold [106] and IPknot [107]. A dataset compiled from PseudoBase [101] consisting of 367 sequences has been used for this work. This dataset has been tested against Rfam [108] to obtain the multiple alignments required by RNAalifold and the sequences for which alignments are available have been selected. After excluding the redundant sequences, the dataset used for the comparative evaluation includes 86 sequences whose lengths are less than 100 nucleotides. The RNA sequences used for the analysis are taken from PseudoBase and contain at least one pseudoknot. The method RNAalifold takes sequence alignments as input to predict the RNA secondary structure. Similar methods which use multiple RNA homologs to compute the RNA secondary structure are accurate but they consume enormous computational resources. For example, as the number of the input RNA sequences increases, the complexity increases exponentially [109]. From Table 4.1 it can be seen that the proposed method has the highest value of correlation coefficient (CC) for all the sequences tested making it a reliable

Table 4.1: Comparison of Different Prediction Methods

Methods	**Performance Criteria**		
	Sensitivity S_n	**Specificity** S_p	**Correlation Coefficient** CC
RNAalifold	0.8330	0.7404	0.5649
IPknot	0.8798	0.6995	0.6176
Hotknots	0.9543	0.6516	0.6212
Dotknot	0.8779	0.7294	0.6137
Proposed Approach	**0.9824**	**0.7222**	**0.6444**

and accurate method.

The proposed approach involves convolution of the matrices B_u and D. So the computational complexity would be similar to that of a 2D convolution which is $O(N^2M^2)$. Here, N is the size of matrix B_u and M is the size of D. As can be seen in Figure 4.3, the base pairing matrix contains mostly 'zeros'. In fact, the only elements in the base pairing matrix containing the non-zero elements are the 'ones' corresponding to the base pairs. Moreover, we are convoluting the upper triangular matrix of B with the diagonal matrix D and without zero padding. This keeps the size of the convolution output C same as the size of B. Hence, the computational complexity would be much less than $O(N^2M^2)$ and depends on the number of base-pairs present in the structure. The proposed approach does not attempt to solve the NP-hardness of the problem. It tries to simplify the problem by considering RNA sequences of short lengths of the range less than 100 base-pairs ($N < 100$). On an average the time taken to process a sequence of length 100 base pairs is 0.28 seconds on a P4, 2.83 GHz computer having 4GB RAM. Online webservers of the methods Hotknots [110], Dotknot [111], RNAalifold [112] and IPknot [113] have been used to compare the results with the proposed method. This is the reason the time complexity of the methods is not included. But, Table 1 in [107] gives some details of the time taken by various methods. The main advantages of the proposed approach is the presence of a pseudoknot in a short RNA sequence is detected fast. Hence, the

proposed approach can be used at places where sorting of RNA sequences is required based on structural similarity. One of the short comings of the approach is failure to detect the non-canonical base-pair between G and U. Moreover, the algorithm tends to maximize the stem patterns which are counter effective in some RNA sequences. Methods based on simultaneous maximization of stem patterns and minimization of the free energy of RNA molecule need to be explored.

4.5 Summary

In this chapter, a reliable and efficient method for predicting the RNA's secondary structure that also includes pseudoknots has been proposed. Prediction of the secondary structure of an RNA sequence involves prediction of the number of stems, the number of loops and the number of pseudoknots, and their corresponding locations. A matched filtering technique has been employed to find the long stem patterns and the corresponding loops in the base pairing matrix of the RNA. This has been achieved by convolving the base paring matrix of the RNA with a diagonal mask of varying size which represents a stem pattern. Once the stems have been identified, this knowledge is then utilized to determine the locations of loops and the presence of pseudoknots. The proposed method has been shown to be computationally efficient as it involves convolution of matrices whose elements are mostly zeros and ones. The proposed method determines the presence of a pseudoknot in an RNA sequence more successfully as compared to other methods such as Hotknots, Dotknot, RNAalifold and IPknot. Experimental results demonstrate the effectiveness and ease of the proposed approach. A graphical tool has also been developed implementing the proposed algorithm to display the secondary structure of an RNA.

Chapter 5

Analysis of Protein Sequences

5.1 Introduction

Proteins are a class of macromolecules synthesized from RNA by the process called translation. Proteins are responsible for carrying out most of the cellular activities. It is interesting to note that cells are made up largely of proteins, such as structural proteins that give the cell rigidity and mobility, proteins that form pores in the cell membrane to control the traffic of small molecules into and out of the cell, and receptor proteins that regulate cellular activities. Proteins are also responsible for most of the metabolic activities of cells. They are essential for the synthesis and breakdown of organic molecules, and for generating the chemical energy needed for cellular activities.

In Chapter 2, it was mentioned that the protein sequences are long chains of amino acids (also referred to as residues) joined by peptide bonds. Due to this reason, proteins are sometimes also referred to as polypeptide chains. Proteins have a tendency to fold into three dimensional (3D) structures, which in turn influence their functionality [9]. The process of *protein folding* is very complex, in which a polypeptide chain

attains a stable 3D structure through short and long range chemical interactions between amino acids which are nearby and in different parts of the molecule, respectively. During this folding process, the polypeptide chain twists and bends until it achieves a state of minimum energy that maximizes the stability of the resulting structure. The three levels of protein structure are shown in Figure 5.1. The primary structure of a protein is the sequence of amino acids present in it. The secondary structure of a protein gives information about the locations of amino acids forming one of the three substructures: α-helix, β-sheet, and loop. Finally, the tertiary structure refers to the three-dimensional arrangement of these substructures forming a complex convoluted protein molecule. For a particular polypeptide, there are many short and long range interactions resulting in several possible folded conformations. A reliable prediction of protein folding is a major challenge.

By virtue of its 3D structure, proteins perform various cellular processes by chemically interacting with other cellular constituents, called targets. These chemical interactions are very specific in nature and occur at specific locations, known as active sites, in the 3D structures. These active sites have particular shapes so that they can fit into the target molecules during their interaction. In and around these active sites are subregions known as *hot-spots* that are responsible for both the chemical stability of active sites as well as supplying the binding energy for the protein-target interactions. A hot-spot may consist of one or more amino acids arranged in an unique pattern in the protein sequence. As the hot-spots play an important role in enabling proteins to perform their functions, a thorough knowledge about their locations is essential for understanding protein function. Therefore, reliable and efficient techniques for identifying the locations of hot-spots in proteins are necessary.

In this chapter, protein sequences are investigated to solve the problems of predicting the protein secondary structure and identifying the locations of hot-spots. For solving the problem of predicting protein secondary structure, a two-stage neural

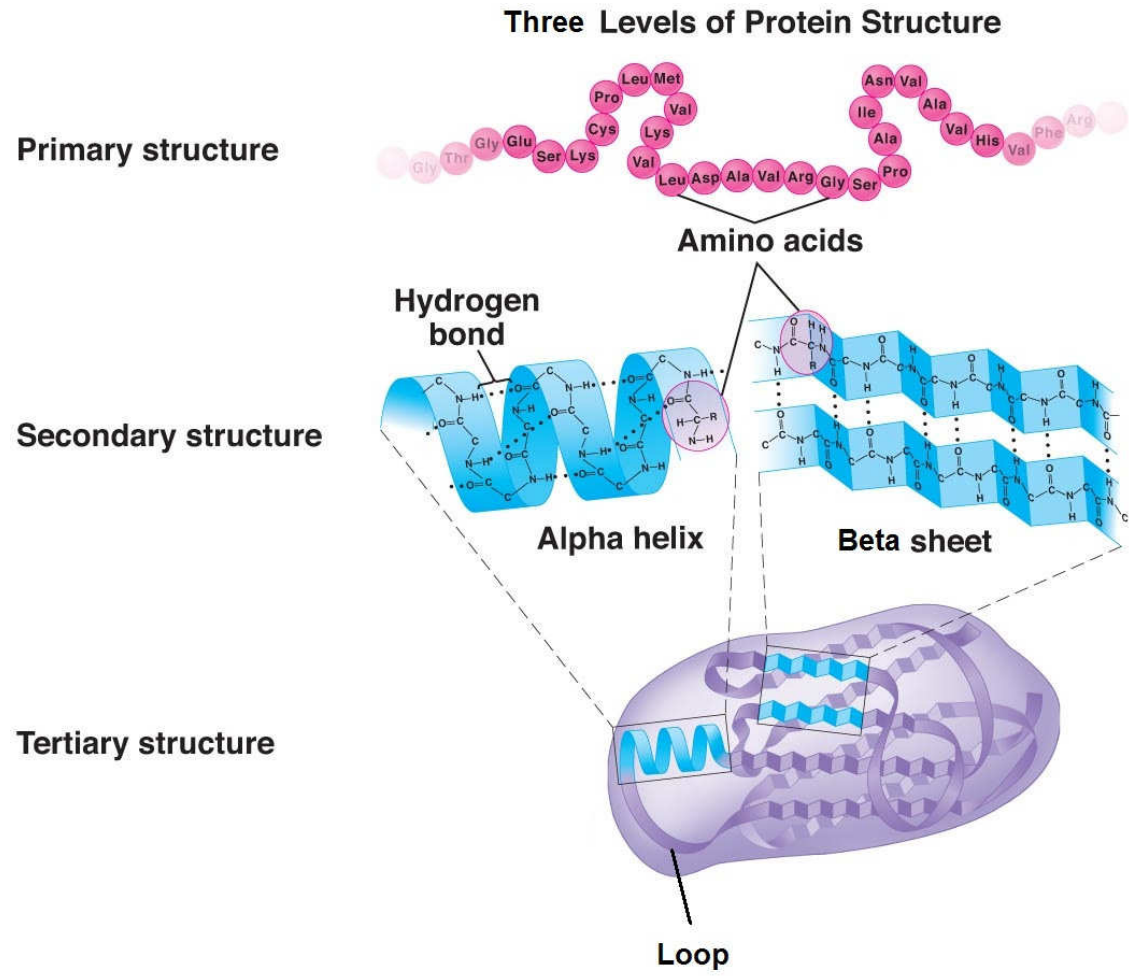


Figure 5.1: Protein secondary structure containing α-helix, β-sheet, and a loop (Source [68]).

network based technique is proposed [114]. The second problem of predicting the locations of hot-spots in protein sequences is investigated using statistically optimal null filters [115].

5.2 Prediction of Protein Secondary Structure

Experimental techniques, such as, x-ray crystallography and nuclear magnetic resonance spectroscopy can provide high resolution structural information of proteins. Unfortunately, these methods are expensive, tedious, time-consuming, and at times

inaccessible. Also, the enormous growth of protein databases (e.g., due to large-scale genome sequencing projects) continues to increase the number of unknown protein sequence-structure pairs. In this scenario, efficient computational techniques, which aid biologists in protein structure prediction, are in great demand. The determination of 3D structure of a protein using computational techniques is rather complicated. Instead, a common practice is to first predict the secondary structure, which can ultimately help determine the 3D structure. Prediction of secondary structure of a protein includes associating each of the amino acids in it to one of the three substructures: α-helix, β-sheet, and loop.

In the last couple of decades, several techniques for protein secondary structure prediction have been proposed as described in Section 1.2. Most of the existing structure prediction methods use a complicated scheme of input encoding to incorporate the evolutionary information. Moreover, the enormous growth of protein databases necessitates the existing prediction models to be extended using huge amounts of training data and developing large-scale neural networks, thereby demanding alternative more efficient modeling techniques. The following section gives the building blocks necessary for prediction of the protein secondary structure using the proposed two-stage neural network models.

5.2.1 Building blocks

Dataset

The development of neural network based models involves the training and validation processes using a suitable data. The protein dataset considered for modeling should be a good representative of the entire protein database. In this work, the widely used RS126 protein dataset developed by Rost and Sander [44] is used for developing the neural models of both the stages. This dataset consists of 126 non-homologous

globular proteins. No two proteins in the dataset have pair-wise sequence similarity greater than 25% for lengths greater than 80 residues. The dataset contains a total of 24,395 amino acids with 32% α-helices, 21% β-sheets and 47% loops.

Encoding scheme for inputs and outputs

The utilization of the neural modeling techniques for the protein prediction problem requires appropriate encoding of input and output data. The secondary structure formed by a residue in a protein sequence is influenced by its neighbors. Therefore, a window approach is adopted to generate the input data. Specifically, the secondary structure formed by the central element or j^{th} residue, R_j, is predicted from a window of amino acids R_{j-n}, ..., R_j,..., R_{j+n}, where the window size, W, is $2n+1$. Usually, n is chosen to be 6, leading to window size, $W = 13$. Each of the residues in the input windowed sequence is encoded using 5-bits. Where as, each of the secondary structure prediction model outputs corresponding to the three substructures, is encoded using 3 bits.

Prediction based on sequence profiles

In this section, an alternative encoding scheme, based on sequence profiles generated is briefly described. The multiple sequence alignments, inferring protein homology, contain additional information about the protein structure. Hence, the use of the multiple sequence alignment information of a given protein, as input to the prediction model, considerably increases the accuracy of secondary structure prediction [44]. Such an approach requires the frequencies of occurrence of all 20 amino acids for each alignment position to be used as the input to the model. Each of these residue frequencies is represented by 3 bits. Further, the N- and C- terminals of the protein are encoded using 3 bits. In essence, a single residue position is encoded using 63 bits ($20 \times 3 + 3$), which in the case of 13 residues of the windowed sub-sequence

translates to an prohibitive number of input neurons (i.e., $13 \times 63 = 819$) making the model training challenging. This complicated encoding scheme is replaced in the proposed technique by the neural model of first stage which is trained to identify the corresponding bin of the input protein. This enables us to use the simple encoding scheme as discussed in the subsection above i.e., the single input sequence is used as input instead of all its homologues.

Neural network models

In the proposed two-stage modeling technique, fully-connected multilayer perceptrons (MLP) neural network models with one hidden layer are used in both the stages. Each node in a layer is connected to all the nodes in the next layer by links associated with real-valued weight parameters. These parameters are first initialized, and then updated during the training process. Backpropagation algorithm, which involves two algorithm-specific parameters, i.e., learning rate and momentum, is used for training the neural models. During training, the k^{th} weight parameter w_k is updated as

$$\Delta w_k(i) = -\gamma \frac{\partial E}{\partial w_k} + h\Delta w_k(i-1), \tag{5.1}$$

where, E is the model error, γ is the learning rate, h is the momentum, and i represents i^{th} iteration/update.

Accuracy measures

A meaningful accuracy measure is critical for evaluating the quality of the models. One of the widely used accuracy measures for secondary structure prediction is given by

$$Q_3 = [\frac{P_\alpha + P_\beta + P_{loop}}{N}] \times 100 \tag{5.2}$$

where P_α, P_β and P_{loop} are the number of correctly predicted α-helices, β-sheets, and loops respectively, and N is total number of residues in a given protein sequence. This measure is also known as three-state overall residue accuracy. Another widely used accuracy measure is the Matthew's correlation coefficient, which in the case of α-helix is defined as

$$C_\alpha = \frac{(p_\alpha n_\alpha) - (u_\alpha o_\alpha)}{\sqrt{(n_\alpha + u_\alpha)(n_\alpha + o_\alpha)(p_\alpha + u_\alpha)(p_\alpha + o_\alpha)}} \tag{5.3}$$

where, p_α is the number of correctly predicted positive cases, n_α is the number of correctly rejected negative cases, o_α is the number of over-predicted cases (false positives), and u_α is the number of under-predicted cases (misses). Coefficients C_α and C_loop can be defined for β-sheet and loop respectively. The coefficients equal 1.0 if the model predictions are 100% correct, equal -1.0 if the predictions are 100% incorrect.

5.3 Proposed Two-stage NN Based Technique

In this section, the proposed two-stage neural network (NN) based technique for protein secondary structure prediction is discussed. The homology information of the input protein can lead to better prediction accuracies of the protein's secondary structure. The 126 non-homologous protein sequences in the RS126 dataset are allotted to 126 different bins. Each of the bins is then populated with the corresponding homologous protein sequences. In other words, all the protein sequences in a bin are homologous and exhibit structural similarities. By doing this, we decompose the structure prediction problem into two tasks. Given a windowed protein sub-sequence, the first task is to associate the sub-sequence to one of the 126 bins. Correspondingly, the first stage of the proposed technique involves development of a neural network model, which can perform this task of associating the input to its corresponding bin.

Having obtained the bin ID of the sequence from the first stage, the second task is to predict the secondary structure. Correspondingly, the second stage of the proposed technique involves development of a neural network model for each of the 126 bins. The output of the second stage neural model is the secondary structure formed by the central residue of the input windowed sub-sequence. In the following sub-sections, we describe the implementation of the two stages.

5.3.1 First stage

The objective of the first stage is to develop a neural network model for bin identification. Input to this model is the windowed protein sub-sequence, and the output is the bin ID to be identified based on its homology. Considering one of the 126 proteins at a time, all its homologues (containing evolutionary information based on multiple sequence alignment) are obtained using PSI-BLAST [116] and are placed in the corresponding bin. By doing so, 126 distinct bins each containing proteins sharing structural similarity are generated. Each bin is assigned a distinct ID, which is encoded as a 7-bit binary number (since $2^7 = 128$). We then divide the protein sequences in each bin into two sets, namely, the training data and the validation data. This completes the preparation of training data.

The next step is to select a neural network to learn the task of bin identification. It is to be noted that the first stage neural network is to be trained using the windowed sub-sequences of proteins (in the training set) as inputs and their corresponding bin IDs as outputs. In this work, the length of the window is set to be 13. Each of the 13 residues is encoded using a 5-bit binary number, since each residue can be one of the 20 amino acids (i.e. $2^5 = 32$). As such, the neural network is selected to have 65 (i.e., 13×5) input neurons. Considering that the bin ID is a 7-bit binary number, the network requires 7 output neurons. Backpropagation option in NeuroModeler [117] is used for training the neural network.

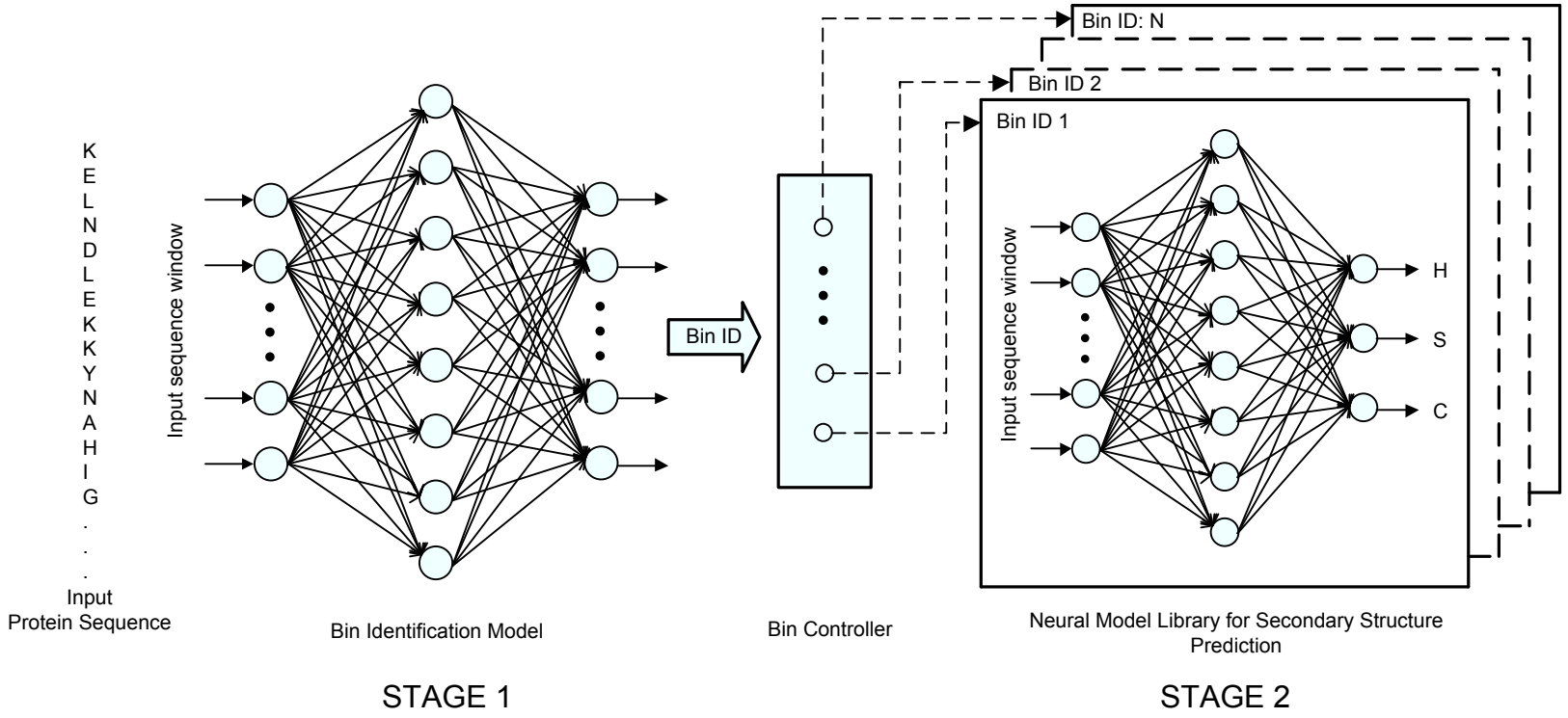


Figure 5.2: Conceptual diagram of the proposed two-stage technique for protein secondary structure prediction

The bin ID predicted by the first-stage neural network is used to select the corresponding neural model from one of the 126 neural models developed for the second stage (see Figure 5.2). It is to be noted that the 65 input and 7 output neural network involves a relatively simpler training process as compared to the standard 819 input and 3 output network described in Section 5.2.1.

5.3.2 Second stage

The objective of the second stage is to develop a set of neural models, where each neural model corresponds to one of the 126 bins. Input to the neural network is the windowed protein sequence, while the output is the secondary structure of the central residue of the windowed sequence. We obtained the structural information of the input protein sequences from the DSSP [118] standard. The next step is to train several neural networks to learn the aforementioned data. Since the input to such neural models is the windowed protein sub-sequence of length 13, each neural network has 65 input neurons similar to the neural network in the first stage. As the output of each of these neural models (secondary structure) is encoded using a 3-bit binary number, the number of output neurons is set to be 3. A total of 126 neural networks are trained using the Backpropagation option in NeuroModeler. After training, the resulting neural models are tested using validation data.

5.3.3 Model utilization

At the end of the two stages described earlier, we have neural models for bin identification and structure prediction. Given a protein sequence, a windowed sub-sequence of length 13 is fed as input to the bin identification neural model, which outputs a bin ID. The same sub-sequence is then used as input to the corresponding neural structure prediction model, which outputs the protein secondary structure. To be

able to predict the structure of the entire protein sequence under consideration, the window needs to be shifted along the protein sequence. In other words, the two-stage neural models are used L times, where L is the length of the protein sequence. In our implementation, zeros are added to the positions of the window where the residues are absent, in order to obtain a window positioning each of the amino acids as the central element.

5.4 Results and Discussion

The proposed two-stage secondary structure prediction model is implemented using the NeuroModeler. The obtained model is trained and validated using the RS126 dataset. The accuracy of the predictions for the validation set obtained using the proposed method is compared with that of the standard PHD method. Both the three-state overall residue accuracy (Q_3) and the Matthew's correlation coefficients are used to assess the performance of each method. A summary of the comparison of both techniques is presented in Table. 5.1. The overall three-state residue accuracy of the proposed technique is 73.4 % using seven-fold cross-validation, which is higher than the standard PHD technique. Considering the complexity of the protein structure prediction problem, this is a considerable improvement. The proposed approach mainly highlights the advantage of binning as compared to the neural model input scheme in the conventional methods. The, Matthew's correlation coefficients obtained suggest that the secondary structure, β-sheet is predicted with less accuracy as compared with the other two secondary structures.

It should be noted that additional bins can be easily incorporated in the proposed approach, expanding the neural model to accommodate more divergent protein sequences. The neural structure prediction models of the second stage are compact and hence easy-to-manage using a bin controller as shown in Figure 5.2. However, it

Table 5.1: Comparison of Protein Secondary Structure Prediction

Method	**Three-state Accuracy**	**Correlation Coefficients**		
	Q_3	C_α	C_β	C_{loop}
PHD	70.8%	0.58	0.50	0.50
Proposed Method	73.4%	0.61	0.49	0.52

is important to appreciate that the first stage neural model needs to be as accurate as possible, since errors in this model propagate to the second stage. A practical difficulty could be that some of the bins might contain inadequate/limited protein sequences, thereby making the neural network training challenging.

5.5 Prediction of Hot-Spots in Proteins

As previously mentioned, proteins are long chains of amino acids, also referred to as residues, joined by peptide bonds. These protein sequences have a tendency to fold into three dimensional (3D) structures, which in turn influence the protein function [119]. Proteins function through interacting with other molecules called *targets* and the *active sites* in proteins aid their interaction with targets. The active sites apart from lending a stable structural configuration to the protein sequence, they also help fitting into specific regions of target molecules thereby facilitating the chemical interaction. Figure 5.3 shows the interaction interface of protein A and protein B. The group of amino acids at this interaction interface are called hot-spots. It is well established that the hot-spots exhibit a characteristic frequency corresponding to their function. Knowing the characteristic frequency of a particular hot-spot, new similar hot-spots can be predicted in other unannotated protein sequences.

There are a number of computational techniques based on digital signal processing proposed in the literature for predicting hot-spots in proteins. More recently, transform techniques such as short-time discrete Fourier transform (STDFT) [66]

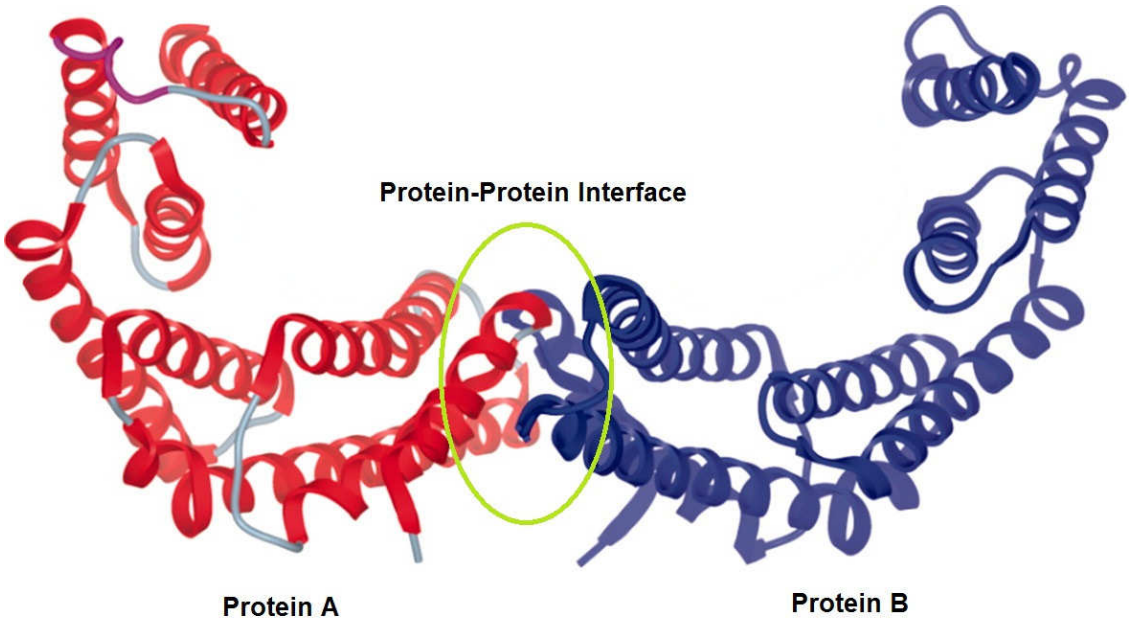


Figure 5.3: Protein-protein interaction.

and modified Morlet continuous-wavelet transform [54] have become popular. Unfortunately, these methods are not entirely reliable, especially the modified Morlet continuous-wavelet transform technique produces more false positives. Moreover, the enormous growth of protein databases (e.g. due to large-scale genome sequencing projects) continues to increase the number of unknown protein sequences to be analyzed for hot-spots. Therefore, to unravel the protein functionality, there is a great need for computational techniques which are more accurate and reliable in locating the hot-spots in proteins.

5.5.1 Related work

The following sections give a brief description of the components required to solve the problem of hot-spot detection in protein sequences. One of the popular methods for hot-spot detection using modified Morlet continuous-wavelet transform is also given. In this work, electron-ion interaction potential (EIIP) values of amino acids

(Table 2.4) are used to map the alphabetical protein sequence to a numerical sequence. EIIP values are physical quantities denoting average energy of valence electrons in the amino acids [67].

Consensus spectrum

In [120], it is observed that a set of protein sequences sharing a common biological function also share a common characteristic frequency. For example, consider a set of M protein sequences sharing a common biological function. Now, the magnitude of the product of the Fourier transforms associated with the numerical sequences of these proteins is defined as

$$P(e^{j\omega}) = \left|X_1(e^{j\omega})X_2(e^{j\omega})\dots X_M(e^{j\omega})\right| \tag{5.4}$$

where $X_1(e^{j\omega})X_2(e^{j\omega})\dots X_M(e^{j\omega})$ are the discrete Fourier transforms corresponding to M proteins respectively. The multiple cross spectrum $P(e^{j\omega})$, also referred to as the *consensus spectrum*, is observed to reveal an interesting feature about the biological function that is common to this set of M proteins and has a distinct peak at the characteristic frequency. As an example, the consensus spectrum of pRb tumor suppressor proteins is shown in Figure 5.4 [121].

Modified Morlet Continuous-Wavelet technique

The modified Morlet continuous-wavelet technique [54] uses the function given by

$$\psi(t) = exp(-\frac{t^2}{a})cos(bt) \tag{5.5}$$

where, a and b are two constants. Since the constant a determines the waveform amplitude modulation degree and the constant b determines the center frequency, they are named as amplitude and frequency factors, respectively. This technique optimizes

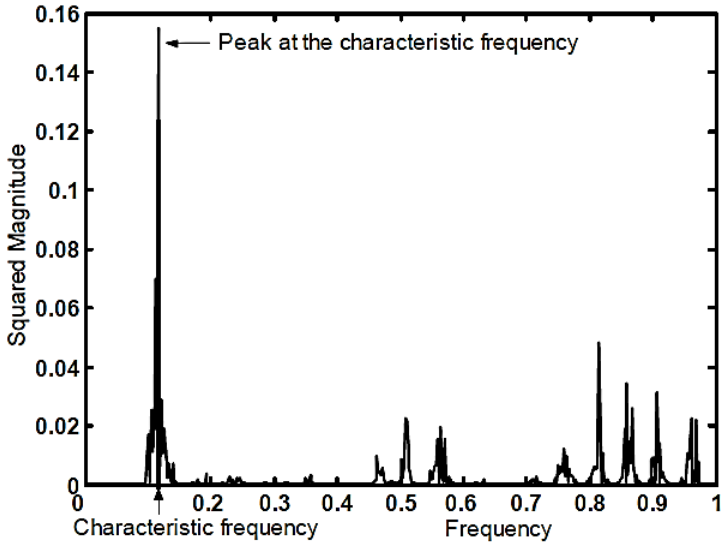


Figure 5.4: Consensus spectrum of pRb proteins. The peak corresponds to characteristic frequency.

the to be applicable for predicting hot-spots in different protein sequences. Optimizing the values of amplitude and frequency factors in the above wavelet function for each and every protein being analyzed makes it computationally inefficient. Moreover, it is shown later that the visual inspection of the high energy regions in the scalogram, produced by the technique, which determine the location of hot-spots can result in a lot of false positives demanding for more accurate and reliable hot-spot prediction techniques.

5.5.2 Proposed SONF based method

In this work, the use of statistically optimally null filters is proposed to solve the problem of hot-spot prediction in protein sequences.

Consider an unannotated protein sequence X, of length N, in which the locations of hot-spots need to be identified. Statistically optimal null filters are utilized in the proposed approach to identify the locations of hot-spots in proteins. In this case, the hot-spots are considered to be the short duration signals (or the message signals) to be located in the DNA sequence X, and the residual signal is the noise. To be able to

feed the sequence X to SONF it is first mapped to an appropriate numerical sequence $X_{EIIP} = \{x_{EIIP}(n)\}$. SONF is a window based approach, and thus a sliding window of length L is used to evaluate if each of the numerical windowed sequences of X_{EIIP}, $X_n = \{x_n(m)\}$, where $n = 1, 2, \ldots, N-L+1$ and $m = n, n+1, \ldots, n+L-1$, contains a hot-spot or not. It can be noted that each of the windowed sequence, X_n, can be expressed as

$$X_n = S_n + R_n \tag{5.6}$$

where $S_n = \{s(m)\}$ is a message signal corresponding to the hot-spot and $R_n = \{r(m)\}$ is a residual signal. S_n and R_n are each of length L. SONF takes the windowed sequence, $X_n = \{x_n(m)\}$, as input and produces the output signal, Y_n, which is an optimal estimate of the message signal S_n. Now, by formulating an appropriate threshold on Y_n, each of the windowed sequence can be classified as belonging to an hot-spot or not.

SONF produces the output Y_n by combining maximum signal-to-noise ratio and least squares optimization criteria. The implementation of the the two-fold optimization in SONF approach is shown in the Figure 2.9, where the instantaneous matched filter (IMF) is first used to detect the presence of a short duration signal embedded in noise by maximizing the signal-to-noise ratio over variable-time observation interval m. The IMF output, I_n, is then scaled by a locally generated function, Λ_n, using least squares (LS) optimization procedure to obtain the optimal estimate, Y_n, of the message signal S_n.

Now, by formulating a binary basis sequence, Φ, according to some characteristic property of the hot-spot, the SONF output, Y_n, can be determined using the recursive relations (2.17). In this case, the initial value of the gain $P(0)$ is chosen to be an identity matrix of order 2, and it is assumed that $\iota(0) = \iota(1)$.

A window size of 35 is utilized for our technique and the window is shifted by one

location. The SNR gain obtained from the ratio of the variance of SONF output to the variance of input signal is plotted against amino acid base index of the protein sequence. Peaks in the resulting plot determine the locations of hot-spots in protein sequences.

5.5.3 Formulation of the basis sequence

A formulation of basis sequence, based on the characteristic frequency of the hot-spot, is very important for identifying them in the input protein sequence. For this purpose a basis sequence containing a set of orthogonal sequences, represented as $\Phi = \{\phi_1, \phi_2\}$, each of which having the characteristic frequency is considered. For example, the basis sequence having the characteristic frequency f can be obtained by using the orthogonal sequences is $\phi_1 = sin(2\pi fnT)$ and $\phi_2 = cos(2\pi fnT)$, where f is the characteristic frequency and T is the period.

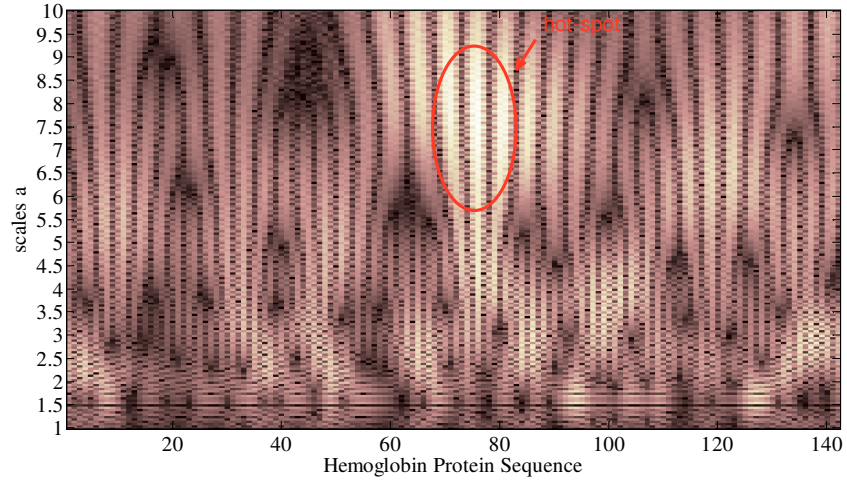
5.6 Results and Discussion

In the proposed SONF based technique, the characteristic frequency which determines the location of hot-spots in protein sequences is modeled as the sinusoidal basis functions. Knowing the characteristic frequency of hot-spots of interest, their presence can be predicted in other annotated protein sequences. The following examples illustrate the effectiveness of the SONF approach over the popular modified Morlet continues-wavelet technique [54].

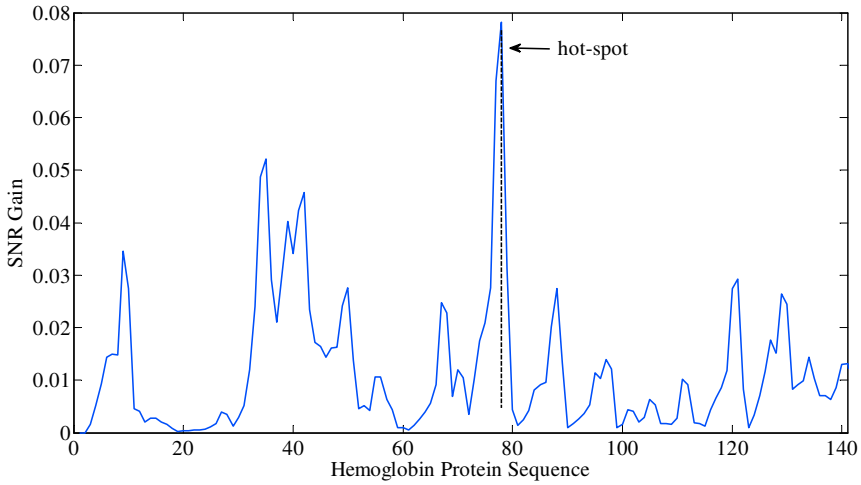
Hemoglobin human α protein active site prediction.

The protein hemoglobin human α has 141 amino acids. The principle function of this protein is to carry oxygen. The modified Morlet continuous-wavelet transform

(CWT) approach is applied on this protein to predict the hot-spots that have affinity to oxygen. The characteristic frequency component of hemoglobin proteins is known to be at $f = 0.0234 \pm 0.008$, as mentioned in [120].



(a)



(b)

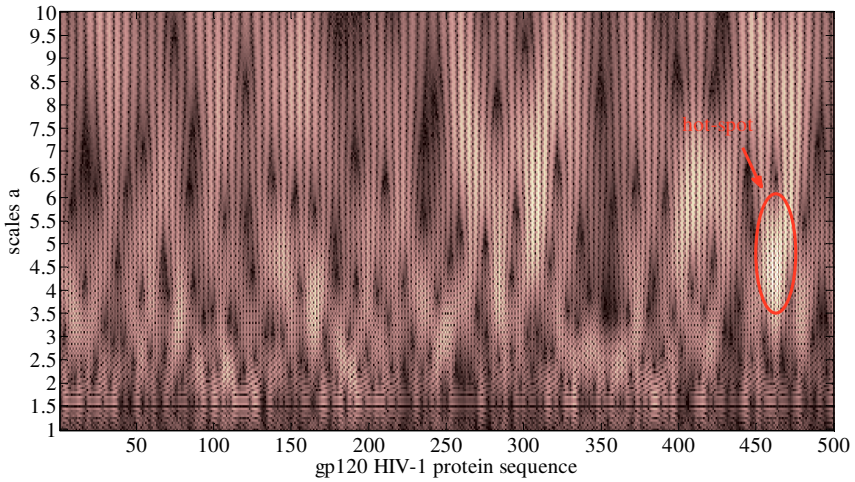
Figure 5.5: Hot-spots in hemoglobin human α protein. (a) Modified Morlet wavelet technique. (b) SONF technique.

The continuous scalogram of this protein obtained by the modified Morlet CWT using an amplitude factor $a = 4$ and a frequency factor of $b = 6$ is shown in Figure 5.5(a). The result of the proposed SONF approach applied to the same hemoglobin

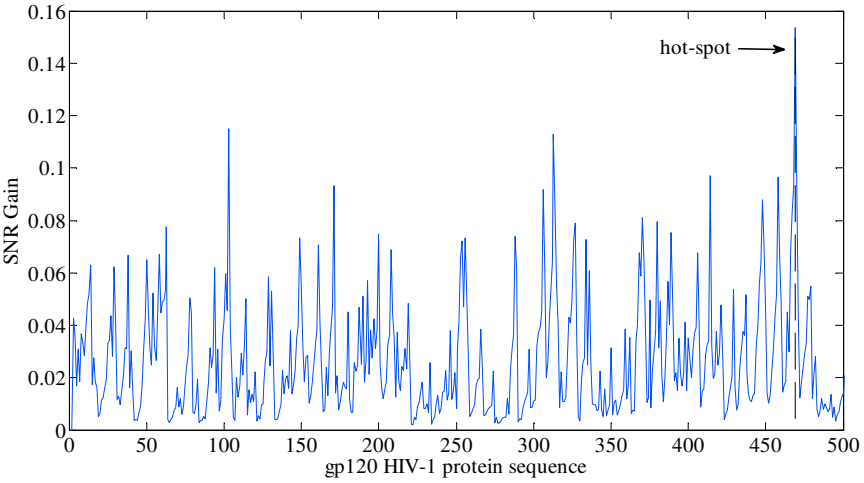
sequence is shown in the Figure 5.5(b). The peak in the plot determines the presence of the hot spot. The basis functions used for the analysis are $sin(2\pi fnT)$ and $cos(2\pi fnT)$ where $f = 0.0234 \pm 0.008$ is the characteristic frequency. It can be seen that its difficult to precisely locate the hot-spot represented by encircled high energy white spot in modified Morlet technique.

HIV envelope protein active site prediction

The infection of host cells by the HIV is due to the interaction between the glycoproteins HIV envelope and the CD4 surface antigen [122]. The characteristic frequency bands assigned to the HIV are 0.06, 0.18, and 0.21 [122]. The modified Morlet technique is applied for the prediction analysis of hot spots of gp120 HIV-1 which is of length 511. The Figure 5.6(a) shows the hot-spot predicted using modified Morlet technique. The modified wavelet with an amplitude factor of 8 and a frequency factor of 5 was used for the gp120 sequence. Again, it is very difficult to accurately locate the bright spot in the scalogram determining the location of the hot-spot. There are other bright spots in Figure 5.6(a) which can be considered as false positives. The result of the proposed SONF approach applied to the gp120 HIV-1 sequence is shown in the Figure 5.6(b). The peak in the plot determines the location of the hot-spot. The peak in the plot determine the location of hot-spots. The above two examples show that the proposed SONF based technique is more reliable in comparison with the modified Morlet technique as it involves optimization of the wavelet parameters for every protein being analyzed and the visual inspection of the bright areas in the scalogram for the location of hot spots is difficult.



(a)



(b)

Figure 5.6: Hot-spots in gp120 HIV-1 protein sequence. (a) Modified Morlet wavelet technique. (b) SONF technique.

5.7 Summary

In this chapter, protein sequences have been analyzed in order to investigate the problems of predicting the protein secondary structures and identifying the locations of hot-spots.

A two-stage neural network based scheme for the prediction of protein secondary structure has been proposed. In the first stage, a neural network has been trained to be able to associate a given input protein sequence to one of the several bins. In the second stage, the corresponding neural network trained for the bin identified in the first stage has been used to predict the protein structure. The proposed two-stage neural network model incorporates additional structural information obtained by the homologues of a protein in order to predict its secondary structure. The RS126 database has been used to validate the prediction results obtained using the proposed method. The proposed two-stage neural network based scheme has been shown to be more effective in accurately predicting the protein secondary structures in comparison to the standard PHD technique.

For predicting the locations of hot-spots in protein sequences, an SONF based approach has been proposed. Hot-spots in a protein exhibit a characteristic frequency corresponding to its functionality. In order to identify hot-spots having a particular functionality, a basis function has been formulated using the characteristic frequency corresponding to the functionality and then employed in the SONF approach. For the formulation of the basis function, two orthogonal sinusoids, having the characteristic frequency of the hot-spots to be predicted, have been used. The SONF based technique utilizes the maximum signal-to-noise ratio and least-squares optimization criteria to predict the hot-spots in protein sequences. The peaks of the SONF output determine the locations of hot-spots. The prediction results obtained by the proposed SONF based approach have been compared with that obtained by the method using the Morlet wavelets, and have been shown to be more accurate.

Chapter 6

Conclusion

6.1 Concluding Remarks

This study has been concerned with an investigation of the problems related to biological sequence analysis using DSP techniques. For this purpose, some of the problems on the analysis of deoxyribonucleic acid (DNA), ribonucleic acid (RNA) and proteins have been studied. Several methods, based on DSP techniques such as statistically optimal null filters (SONF), matched filters and neural networks, have been developed as a result of this investigation.

In the first part of this study, DNA sequences have been analyzed to identify the locations of CpG islands (CGIs) and protein coding regions (exons). These analyses have been carried out by developing techniques based on an SONF approach. For locating CGIs, a basis function has been formulated and used in SONF, which is implemented by combining the criteria of maximization of signal-to-noise ratio and least square optimization. The performance of the proposed technique for the prediction of CGIs has been tested on four randomly chosen contigs in chromosomes 21 and 22 of human beings. One of the main features of the proposed approach is that it does not

depend on the transition probability tables utilized by some of the existing methods. It has been shown that the use of the basis sequence instead of the transition probability tables, obtained from training data, is more reliable. The prediction accuracy of the proposed approach has been shown to be more than 97%. For predicting the locations of protein coding regions, i.e., exons, a basis function based on the period-3 property has been formulated and used by SONF to predict the locations of exons in DNA sequences. The proposed algorithm has been tested using chromosome III of *C. elegans* and the results have been validated making use of the existing knowledge of annotations of this sequence.

In the second part of this thesis, RNA sequences have been analyzed in order to predict their secondary structures. For this purpose, matched filters based on 2-dimensional convolution have been developed to identify the numbers and locations of stem and loop patterns. The knowledge of the stem and loop patterns thus obtained has been used to predict the presence of pseudoknots, thereby providing the entire RNA secondary structure. The proposed matched filtering based method has been tested using the Pseudobase database. The proposed algorithm is compared with some of the existing methods, and it has been shown to provide a better result in the context of the already known results on the existance of the RNA secondary structure in the sequences of the database. The stem patterns in an RNA structure are manifested as diagonal lines in the dotplot of the RNA. It has been shown that these diagonal lines, representing the stem patterns, can be identified more easily using the proposed matched filtering approach in comparison to that using other techniques such as dynamic programming, thermodynamic energy considerations, etc. A graphical user interface (GUI), which predicts and displays the RNA secondary structure, has also been designed.

Finally, in the last part of this thesis, protein sequences have been analyzed to predict their secondary structures and to identify the locations of the hot-spots. A

two-stage neural network scheme has been proposed for predicting the protein secondary structures. In the first stage, a neural network model is trained to be able to associate the protein sequence to one of the several bins containing its homologues. In the second stage, a neural network trained for the bin identified in the first stage is used to predict the protein structure. The proposed two-stage neural network based scheme has been tested using the RS126 database and its performance compared with that of an existing method, namely PHD. It has been shown that the proposed scheme provides more accurate predictions in terms of the three-state accuracy and correlation performance metrics. The solution to the problem of predicting the hot-spots in proteins has been obtained using the SONF approach. A hot-spot in protein sequence exhibits a characteristic frequency corresponding to its biological function. This frequency has been used to formulate a basis function, which is used in SONF to detect the locations of the hot-spots belonging to the functional group characterized by this frequency. The proposed technique has been compared with that using the Morlet wavelets, and it has been shown to be more accurate in obtaining the locations of the hot-spots.

6.2 Scope for Further Investigation

The DSP based techniques proposed in this thesis for analyzing biological sequences, focuses mainly on predicting motifs such as exons, CGIs and hot-spots. In these techniques, the characteristic properties of these motifs have been used to formulate the basis sequences, and then employed in the SONF approach. Further investigations can be carried out to identify other possible characteristic properties of these motifs with a view to enhancing their prediction. Problems, such as sequence alignment and sequence comparison, could also be investigated using the SONF approach. In these cases, one of the two sequences being compared could be considered as the

input sequence, and the other as the basis sequence. The peaks in the SONF output obtained could then be used for determining the extant of similarity between the sequences [123, 124]. Finally, for predicting the RNA secondary structure, energy considerations could also be incorporated in the proposed matched filtering approach to enhance its prediction performance.

References

[1] F. Crick and J. Watson, "Molecular structure of nucleic acids," *Nature*, vol. 171, no. 4356, pp. 737–738, April 1953.

[2] R. Franklin and R. Gosling, "Evidence for 2-chain helix in crystalline structure of sodium deoxyribonucleate," *Nature*, vol. 172, pp. 156–157, April 1953.

[3] R. Fleischmann *et al.*, "Whole-genome random sequencing and assembly of haemophilus," *Science*, vol. 269, no. 5223, pp. 496–512, July 1995.

[4] GenBank. (2012, Aug.) Genetic sequence database. [Online]. Available: www.ncbi.nlm.nih.gov/genbank/.

[5] NCBI. (2012, Aug.) National center for biotechnology information database. [Online]. Available: http://www.ncbi.nlm.nih.gov/.

[6] DDBJ. (2012, Aug.) DNA data bank of japan. [Online]. Available: www.ddbj.nig.ac.jp/.

[7] EBI. (2012, Aug.) EBI: European bioinformatics institute. [Online]. Available: www.ebi.ac.uk/.

[8] M. Gelfand, "Prediction of function in DNA sequence analysis," *Journal of Computational Biology*, vol. 2, no. 1, pp. 87–115, 1995.

[9] D. Clark and N. Pazdernik, *Molecular biology.* Waltham: Academic Press, 2012.

[10] M. Gardiner-Garden and M. Frommer, "CpG islands in vertebrate genomes." *Journal of Molecular Biology*, vol. 196, no. 2, p. 261, July 1987.

[11] P. Rice *et al.*, "EMBOSS: the European molecular biology open software suite," *Trends in Genetics*, vol. 16, no. 6, pp. 276–277, June 2000.

[12] L. Ponger and D. Mouchiroud, "CpGProD: identifying CpG islands associated with transcription start sites in large genomic mammalian sequences," *Bioinformatics*, vol. 18, no. 4, p. 631, July 2002.

[13] N. Dasgupta, S. Lin, and L. Carin, "Sequential modeling for identifying CpG island locations in human genome," *IEEE Signal Processing Letters*, vol. 9, no. 12, pp. 407–409, March 2002.

[14] P. Luque-Escamilla *et al.*, "Compositional searching of CpG islands in the human genome," *Physical Review E*, vol. 71, no. 6, p. 61925, June 2005.

[15] C. Bock, J. Walter, M. Paulsen, and T. Lengauer, "CpG island mapping by epigenome prediction," *PLoS Comput Biol*, vol. 3, no. 6, p. e110, June 2007.

[16] Y. Sujuan, A. Asaithambi, and Y. Liu, "CpGIF: an algorithm for the identification of CpG islands," *Bioinformation*, vol. 2, no. 8, p. 335, May 2008.

[17] D. Takai and P. Jones, "Comprehensive analysis of CpG islands in human chromosomes 21 and 22," *Proceedings of the National Academy of Sciences*, vol. 99, no. 6, pp. 3740–3745, 1987.

[18] M. Hackenberg *et al.*, "CpGcluster: a distance-based algorithm for CpG-island detection," *BMC Bioinformatics*, vol. 7, no. 1, p. 446, October 2006.

[19] L. Han and Z. Zhao, "CpG islands or CpG clusters: how to identify functional GC-rich regions in a genome?" *BMC Bioinformatics*, vol. 10, no. 1, p. 65, January 2009.

[20] M. Stanke and B. Morgenstern, "AUGUSTUS: a web server for gene prediction in eukaryotes that allows user-defined constraints," *Nucleic Acids Research*, vol. 33, no. suppl 2, pp. W465–W467, March 2005.

[21] E. Blanco, G. Parra, and R. Guigó, "Using geneid to identify genes," *Current Protocols in Bioinformatics*, pp. 4–3, 2007.

[22] C. Burge *et al.*, "Prediction of complete gene structures in human genomic DNA," *Journal of Molecular Biology*, vol. 268, no. 1, pp. 78–94, April 1997.

[23] A. Krogh, "Two methods for improving performance of an HMM and their application for gene finding," in *Proc. Int. Conf. on Intelligent Systems for Molecular Biology*, Menlo Park, 1997, pp. 179–186.

[24] K. Murakami and T. Takagi, "Gene recognition by combination of several gene-finding programs." *Bioinformatics*, vol. 14, no. 8, pp. 665–675, 1998.

[25] V. Pavlović, A. Garg, and S. Kasif, "A bayesian framework for combining gene predictions," *Bioinformatics*, vol. 18, no. 1, pp. 19–27, 2002.

[26] J. Abrahams, M. van den Berg, E. van Batenburg, and C. Pleij, "Prediction of RNA secondary structure, including pseudoknotting, by computer simulation." *Nucleic Acids Research*, vol. 18, no. 10, p. 3035, May 1990.

[27] A. Gultyaev, F. Van Batenburg, and C. Pleij, "The computer simulation of RNA folding pathways using a genetic algorithm," *Journal of Molecular Biology*, vol. 250, no. 1, pp. 37–51, June 1995.

[28] R. Cary and G. Stormo, "Graph-theoretic approach to RNA modeling using comparative data," in *Proc. Intl. Conf. on Intelligent Systems for Molecular Biology*, Cambridge, July 1995, pp. 75–80.

[29] M. Zuker, "Mfold web server for nucleic acid folding and hybridization prediction," *Nucleic Acids Research*, vol. 31, no. 13, p. 3406, July 2003.

[30] M. Zuker and P. Stiegler, "Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information," *Nucleic Acids Research*, vol. 9, no. 1, p. 133, May 1981.

[31] M. Zuker, "Prediction of RNA secondary structure by energy minimization," *Methods Molecular Biology*, vol. 25, pp. 267–294, January 1994.

[32] J. Deogun, R. Donis, O. Komina, and F. Ma, "RNA secondary structure prediction with simple pseudoknots," in *Proceedings of the Second Conference on Asia-Pacific Bioinformatics*, Dunedin, New Zealand, January 2004, pp. 239–246.

[33] D. Mathews and D. Turner, "Dynalign: an algorithm for finding the secondary structure common to two RNA sequences," *Journal of Molecular Biology*, vol. 317, no. 2, pp. 191–203, 2002.

[34] E. Rivas and S. Eddy, "A dynamic programming algorithm for RNA structure prediction including pseudoknots," *Journal of Molecular Biology*, vol. 285, no. 5, pp. 2053–2068, 1999.

[35] J. Reeder, P. Steffen, and R. Giegerich, "pknotsRG: RNA pseudoknot folding including near-optimal structures and sliding windows," *Nucleic Acids Research*, vol. 35, no. suppl 2, pp. W320–W324, July 2007.

[36] E. Rivas and S. Eddy, "The language of RNA: a formal grammar that includes pseudoknots," *Bioinformatics*, vol. 16, no. 4, p. 334, March 2000.

[37] B. Knudsen and J. Hein, "RNA secondary structure prediction using stochastic context-free grammars and evolutionary history." *Bioinformatics*, vol. 15, no. 6, pp. 446–454, June 1999.

[38] R. Dowell and S. Eddy, "Evaluation of several lightweight stochastic context-free grammars for RNA secondary structure prediction," *BMC Bioinformatics*, vol. 5, no. 1, p. 71, August 2004.

[39] Y. Kato, H. Seki, and T. Kasami, "RNA pseudoknotted structure prediction using stochastic multiple context-free grammar," *IPSJ Digital Courier*, vol. 2, no. 0, pp. 655–664, February 2006.

[40] P. Chou and G. Fasman, "Prediction of protein conformation," *Biochemistry*, vol. 13, no. 2, pp. 222–245, 1974.

[41] J. Garnier *et al.*, "GOR method for predicting protein secondary structure from amino acid sequence." *Methods in Enzymology*, vol. 266, p. 540, 1996.

[42] N. Qian and T. Sejnowski, "Predicting the secondary structure of globular proteins using neural network models," *Journal of Molecular Biology*, vol. 202, no. 4, pp. 865–884, August 1988.

[43] B. Rost and C. Sander, "Combining evolutionary information and neural networks to predict protein secondary structure," *Proteins: Structure, Function, and Bioinformatics*, vol. 19, no. 1, pp. 55–72, May 1994.

[44] B. Rost *et al.*, "Prediction of protein secondary structure at better than 70% accuracy," *Journal of Molecular Biology*, vol. 232, pp. 584–584, July 1993.

[45] D. Jones, "Protein secondary structure prediction based on position-specific scoring matrices," *Journal of Molecular Biology*, vol. 292, no. 2, pp. 195–202, September 1999.

[46] S. Hua and Z. Sun, "A novel method of protein secondary structure prediction with high segment overlap measure: support vector machine approach," *Journal of Molecular Biology*, vol. 308, no. 2, pp. 397–407, April 2001.

[47] T. Kortemme, D. Kim, and D. Baker, "Computational alanine scanning of protein-protein interfaces," *Science Signalling*, vol. 2004, no. 219, p. pl2, January 2004.

[48] W. DeLano, "Unraveling hot spots in binding interfaces: progress and challenges," *Current Opinion in Structural Biology*, vol. 12, no. 1, pp. 14–20.

[49] D. Rajamani, S. Thiel, S. Vajda, and C. Camacho, "Anchor residues in protein–protein interactions," *Proc. National Academy of Sciences of the United States of America*, vol. 101, no. 31, pp. 11 287–11 292, August 2004.

[50] D. González-Ruiz and H. Gohlke, "Targeting protein-protein interactions with small molecules: challenges and perspectives for omputational binding epitope detection and ligand finding," *Current Medicinal Chemistry*, vol. 13, no. 22, pp. 2607–2625, December 2006.

[51] D. Anastassiou, "Genomic signal processing," *Signal Processing Magazine, IEEE*, vol. 18, no. 4, pp. 8–20, November 2001.

[52] P. Vaidyanathan and B. Yoon, "The role of signal-processing concepts in genomics and proteomics," *Journal of the Franklin Institute*, vol. 341, no. 1-2, pp. 111–135, March 2004.

[53] S. Datta and A. Asif, "A fast DFT based gene prediction algorithm for identification of protein coding regions," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, vol. 5, March 2005, pp. 653–656.

[54] K. Deergha Rao and M. N. S. Swamy, "Analysis of genomics and proteomics using DSP techniques," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 55, no. 1, pp. 370–378, 2008.

[55] S. Tiwari *et al.*, "Prediction of probable genes by Fourier analysis of genomic sequences," *Bioinformatics*, vol. 13, no. 3, pp. 263–270, June 1997.

[56] P. Vaidyanathan and B. J. Yoon, "Digital filters for gene prediction applications," in *Proc. IEEE Int. Asilomar Conference on Signals, Systems and Computers*, vol. 1, New York, May 2002, pp. 306–310.

[57] N. Song and H. Yan, "Short exon detection in DNA sequences based on multifeature spectral analysis," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, p. 2, March 2011.

[58] J. P. Mena-Chalco, H. Carrer, Y. Zana, and R. M. Cesar, "Identification of protein coding regions using the modified Gabor-wavelet transform," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 5, no. 2, pp. 198–207, June 2008.

[59] J. Tuqan and A. Rushdi, "A DSP approach for finding the codon bias in DNA sequences," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 3, pp. 343–356, June 2008.

[60] R. Guan and J. Tuqan, "Multirate DSP models for gene detection," in *Proc. IEEE Int. Asilomar Conference on Signals, Systems and Computers*, vol. 2, Pacific Grove, California, November 2004, pp. 1641–1645.

[61] J. Darned, H. Lodish, and D. Baltimore, "Molecular cell biology," *The Scientist*, vol. 1, no. 1, p. 20, December 1986.

[62] R. Durbin, *Biological sequence analysis: Probabilistic models of proteins and nucleic acids*. Cambridge: Cambridge University Press, 1998.

[63] B. Yoon and P. Vaidyanathan, "Identification of CpG islands using a bank of IIR lowpass filters," in *Proc. IEEE Int. Digital Signal Processing Workshop*, Austin, Texas, 2004, pp. 315–319.

[64] A. Rushdi and J. Tuqan, "A new DSP-based measure for CPG islands detection," in *Proc. IEEE Int. Digital Signal Processing Workshop*, Banf, Canada, 2006, pp. 561–565.

[65] B. Yoon and P. Vaidyanathan, "Fast structural similarity search of noncoding rnas based on matched filtering of stem patterns," in *Proc. IEEE Int. Asilomar Conference on Signals, Systems and Computers*, 2007, pp. 44–48.

[66] P. Ramachandran, A. Antoniou, and P. Vaidyanathan, "Identification and location of hot spots in proteins using the short-time discrete Fourier transform," in *Proc. IEEE Intl. Signals, Asilomar Conference on Systems and Computers*, vol. 2, 2004, pp. 1656–1660.

[67] P. Ramachandran and A. Antoniou, "Identification of hot-spot locations in proteins using digital filters," *IEEE Journal of Selected Topics in Signal Processing*, vol. 2, no. 3, pp. 378–389, August 2008.

[68] N. Campbell, J. Reece, and M. Taylor, *Biology: Concepts and Connections*. Pearson Education, 2009.

[69] H. Kwan and S. Arniker, "Numerical representation of DNA sequences," in *Proc. IEEE Int. Conf. Electro/Information Technology*, Windsor, Canada, June 2009, pp. 307–310.

[70] R. Voss, "Evolution of long-range fractal correlations and 1/f noise in dna base sequences," *Physical Review Letters*, vol. 68, no. 25, pp. 3805–3808, 1992.

[71] V. Veljkovic, I. Dimitrijevic, and D. Lalovic, "Is it possible to analyze DNA and protein sequences by the methods of digital signal processing?" *IEEE Transactions on Biomedical Engineering*, pp. 337–341, 1985.

[72] R. Agarwal, E. I. Plotkin, and M. N. S. Swamy, "Statistically optimal null filter based on instantaneous matched processing," *Circuits, Systems, and Signal Processing*, vol. 20, no. 1, pp. 37–61, 2001.

[73] R. Yadav, M. N. S. Swamy, and R. Agarwal, "Model-based seizure detection for intracranial EEG recordings," *IEEE Transactions on Biomedical Engineering,*, vol. 59, no. 5, pp. 1419–1428, 2012.

[74] E. I. Plotkin, "Signal-controlled tim-series modeling based on arma blocks, and separation of superimposed, overlapping spectra signals," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. 79, no. 10, pp. 1676–1681, 1996.

[75] M. Burset *et al.*, "Evaluation of gene structure prediction programs," *Genomics*, vol. 34, no. 3, pp. 353–367, June 1996.

[76] F. Antequera and A. Bird, "Number of CpG islands and genes in human and mouse," *Proc. National Academy of Sciences of the United States of America*, vol. 90, no. 24, pp. 11 995–11 999, December 1993.

[77] F. Larsen, G. Gundersen, R. Lopez, and H. Prydz, "CpG islands as gene markers in the human genome," *Genomics*, vol. 13, no. 4, pp. 1095–1107, August 1992.

[78] Y. Wang and F. Leung, "An evaluation of new criteria for CpG islands in the human genome as gene markers," *Bioinformatics*, vol. 20, no. 7, p. 1170, January 2004.

[79] F. Antequera and A. Bird, "CpG islands as genomic footprints of promoters that are associated with replication origins," *Current Biology*, vol. 9, pp. 661–667, October 1999.

[80] I. Ioshikhes and M. Zhang, "Large-scale human promoter mapping using CpG islands," *Nature Genetics*, vol. 26, no. 1, pp. 61–63, September 2000.

[81] F. Antequera, "Structure, function and evolution of CpG island promoters," *Cellular and Molecular Life Sciences*, vol. 60, no. 8, pp. 1647–1658, March 2003.

[82] S. Saxonov, P. Berg, and D. Brutlag, "A genome-wide analysis of cpg dinucleotides in the human genome distinguishes two distinct classes of promoters," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 103, no. 5, pp. 1412–1417, January 2006.

[83] A. Bird, "DNA methylation patterns and epigenetic memory," *Genes & Development*, vol. 16, no. 1, pp. 6–21, January 2002.

[84] J. Herman and S. Baylin, "Gene silencing in cancer in association with promoter hypermethylation," *The New England Journal of Medicine*, vol. 349, no. 21, p. 2042, February 2003.

[85] J. Issa, "CpG island methylator phenotype in cancer," *Nature Reviews Cancer*, vol. 4, no. 12, pp. 988–993, December 2004.

[86] R. Illingworth *et al.*, "A novel CpG island set identifies tissue-specific methylation at developmental gene loci," *PLoS Biol*, vol. 6, no. 1, p. e22, January 2008.

[87] L. Heisler *et al.*, "CpG Island microarray probe sequences derived from a physical library are representative of CpG Islands annotated on the human genome," *Nucleic Acids Research*, vol. 33, no. 9, p. 2952, June 2005.

[88] R. Kakumani, M. O. Ahmad, and V. Devabhaktuni, "Identification of CpG islands in DNA sequences using matched filters," in *Proc. IEEE Int. Conf. Engineering in Medicine and Biology Society (EMBC)*, Boston, Massachusetts, September 2011, pp. 6029–6032.

[89] R. Kakumani, M. O. Ahmad, and V. K. Devabhaktuni, "Identification of CpG islands in DNA sequences using statistically optimal null filters." *EURASIP Journal on Bioinformatics and Systems Biology*, vol. 2012, no. 12, pp. 1–14, August 2012.

[90] R. Kakumani, V. Devabhaktuni, and M. Omair Ahmad, "Prediction of protein-coding regions in DNA sequences using a model-based approach," in *Proc. IEEE*

Int. Symposium on Circuits and Systems (ISCAS), Seattle, Washington, May 2008, pp. 1918–1921.

[91] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, February 1989.

[92] B. Liu, *Statistical genomics: linkage, mapping, and QTL analysis*. CRC Press, 1998.

[93] E. Trifonov and J. Sussman, "The pitch of chromatin DNA is reflected in its nucleotide sequence," *Proceedings of the National Academy of Sciences*, vol. 77, no. 7, pp. 3816–3820, October 1980.

[94] A. V. Oppenheim *et al.*, *Discrete-time signal processing*. New Jersey: Prentice hall Englewood Cliffs, 1989.

[95] P. Vaidyanathan and B.-J. Yoon, "Gene and exon prediction using allpass-based filters," in *Workshop on Genomic Sig. Proc. And Stat.*, Raleigh, NC.

[96] J. Mattick, "The functional genomics of noncoding RNA," *Science*, vol. 309, no. 5740, p. 1527, 2005.

[97] V. Bafna and S. Zhang, "FastR: fast database search tool for non-coding RNA," in *Proc. IEEE Int. Computational Systems Bioinformatics Conference*, Miami, Florida, August 2004, pp. 52–61.

[98] G. Storz, "An expanding universe of noncoding RNAs," *Science*, vol. 296, no. 5571, p. 1260, April 2002.

[99] R. Kakumani, M. O. Ahmad, and V. K. Devabhaktuni, "Prediction of secondary structure of RNAs with pseudoknots using matched filters," *Journal of Biological Systems*, vol. 20, no. 4, pp. 455–469, December 2012.

[100] R. Kakumani, M. Omair Ahmad, and V. Devabhaktuni, "Prediction of secondary structure of RNAs with pseudoknots using matched filter," in *Proc. IEEE Int. Circuits and Systems (NEWCAS)*, Montreal, Canada, June 2010, pp. 9–12.

[101] PseudoBase. (2012, Aug.) Pseudoknot database. [Online]. Available: http://www.ekevanbatenburg.nl/PKBASE/PKB.HTML.

[102] F. Van Batenburg, A. Gultyaev, and C. Pleij, "PseudoBase: structural information on RNA pseudoknots," *Nucleic Acids Research*, vol. 29, no. 1, p. 194, January 2001.

[103] VARNA. (2012, Aug.) RNA structure visualization tool. [Online]. Available: http://varna.lri.fr/.

[104] J. Ren, B. Rastegari, A. Condon, and H. Hoos, "HotKnots: heuristic prediction of RNA secondary structures including pseudoknots," *RNA*, vol. 11, no. 10, pp. 1494–1504, October 2005.

[105] J. Sperschneider and A. Datta, "DotKnot: pseudoknot prediction using the probability dot plot under a refined energy model," *Nucleic Acids Research*, vol. 38, no. 7, pp. e103–e103, January 2010.

[106] S. Bernhart, I. Hofacker, S. Will, A. Gruber, and P. Stadler, "RNAalifold: improved consensus structure prediction for RNA alignments," *BMC Bioinformatics*, vol. 9, no. 1, p. 474, February 2008.

[107] K. Sato, Y. Kato, M. Hamada, T. Akutsu, and K. Asai, "IPknot: fast and accurate prediction of RNA secondary structures with pseudoknots using integer programming," *Bioinformatics*, vol. 27, no. 13, pp. i85–i93, July 2011.

[108] P. Gardner, J. Daub, J. Tate, B. Moore, I. Osuch, S. Griffiths-Jones, R. Finn, E. Nawrocki, D. Kolbe, S. Eddy *et al.*, "Rfam: Wikipedia, clans and the decimal release," *Nucleic Acids Research*, vol. 39, no. suppl 1, pp. D141–D145, January 2011.

[109] A. Harmanci, G. Sharma, and D. Mathews, "TurboFold: iterative probabilistic estimation of secondary structures for multiple RNA sequences," *BMC Bioinformatics*, vol. 12, no. 1, p. 108, November 2011.

[110] HotKnots. (2012, Aug.) RNA secondary structure prediction tool. [Online]. Available: http://www.rnasoft.ca/cgi-bin/RNAsoft/HotKnots/hotknots.pl.

[111] DotKnot. (2012, Aug.) RNA secondary structure prediction tool. [Online]. Available: http://dotknot.csse.uwa.edu.au/.

[112] RNAalifold. (2012, Aug.) RNA secondary structure prediction tool. [Online]. Available: http://rna.tbi.univie.ac.at/cgi-bin/RNAalifold.cgi.

[113] IPknot. (2012, Aug.) RNA secondary structure prediction tool. [Online]. Available: http://rna.naist.jp/ipknot/.

[114] R. Kakumani, V. Devabhaktuni, and M. O. Ahmad, "A two-stage neural network based technique for protein secondary structure prediction," in *Proc. IEEE Int. Conf. Engineering in Medicine and Biology Society (EMBS)*, Vancouer, Canada, August 2008, pp. 1355–1358.

[115] R. Kakumani, M. O. Ahmad, and V. Devabhaktuni, "Prediction of hot-spots in protein sequences using statistically optimal null filters," in *Proc. IEEE Int. Conf. Circuits and Systems Conference (NEWCAS)*, Montreal, Canada, June 2012, pp. 121–124.

[116] A. A Schäffer *et al.*, "IMPALA: matching a protein sequence against a collection of psi-blast-constructed position-specific score matrices," *Bioinformatics*, vol. 15, no. 12, pp. 1000–1011, January 1999.

[117] NeuroModeler. (2007, Aug.) A RF/microwave oriented software tool. [Online]. Available: http://neuroweb.doe.carleton.ca/main.html.

[118] W. Kabsch and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers*, vol. 22, no. 12, pp. 2577–2637, December 1983.

[119] C. Branden *et al.*, *Introduction to protein structure.* New York: Garland New York., 1991.

[120] I. Cosic, "Macromolecular bioactivity: Is it resonant interaction between macromolecules? - Theory and applications," *IEEE Transactions on Biomedical Engineering*, vol. 41, no. 12, pp. 1101–1114, April 1994.

[121] I. Cosic, E. Pirogova, and M. Akay, "Application of the resonant recognition model to analysis of interaction between viral and tumor suppressor proteins," in *Proc. IEEE Intl. Conf. Engineering in Medicine and Biology Society (EMBS)*, vol. 3, Cancun, Mexico, August 2003, pp. 2398–2401.

[122] I. Cosic, "Analysis of HIV proteins using DSP techniques," in *Proc. IEEE Intl. Conf. Engineering in Medicine and Biology Society (EMBS)*, vol. 3, 2001, pp. 2886–2889.

[123] K. Rajasekhar, M. O. Ahmad, and V. Devabhaktuni, "Finding specific RNA sequence motifs using digital filters," in *Proc. IEEE Int. Conf. Canadian Conference on Electrical & Computer Engineering (CCECE)*, Montreal, Canada, May 2012, pp. 1–4.

[124] R. Kakumani, M. O. Ahmad, and V. Devabhaktuni, "Comparative genomic analysis using statistically optimal null filters," in *Proc. IEEE Int. Symposium on Circuits and Systems (ISCAS)*, Paris, France, June 2010, pp. 2235–2238.