Adaptive computing for In Silico recognition of hormone response elements

Stepanova, Maria

2008

Stepanova, M. (2008). Adaptive computing for In Silico recognition of hormone response elements. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/14572>

<https://doi.org/10.32657/10356/14572>

ADAPTIVE COMPUTING FOR *IN SILICO* RECOGNITION OF HORMONE RESPONSE ELEMENTS

MARIA STEPANOVA

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirement for the degree of
Doctor of Philosophy

2008

Acknowledgements

First of all, I would like to express my sincere gratitude and appreciation to my research supervisor Prof. Lin Feng who have led me through the pitfalls of obtaining a high degree. He was giving continuous advice and excellent comments that enabled me to complete the thesis successfully. Without his guidance, support, encouragement, and patience when supervising and discussing my work, I might never be able to finish the research and writing. I also gratefully appreciate his valuable time that he spent reading and criticizing my papers and the impact he has made to the final version of my thesis.

I am also very grateful to my co-supervisor Prof. Valerie Lin from SBS. Without her valuable advices and excellent explanations, I would hardly be able not to get lost in the diversity of molecular biology and endocrinology knowledge. She was always kind and helpful assisting me to interpret my results and advising how to make my efforts useful for real life.

I would like to express my gratitude to the SCE professors Prof Jagath Rajapakse and Prof Bertil Schmidt who have taught very useful courses which helped me a lot to advance in the area of bioinformatics.

I also wish to express my appreciation to all staff members and students of BioInformatics Research Center who provided a pleasant atmosphere during the entire period my PhD program. I am especially grateful to BIRC students Alice Cheong and Jesslyn Saw for valuable discussions of various aspects of our research.

I wish to thank my husband Hennadiy for his continuous support during hard time of my work and life. Without his care I would have lost motivation dozens of times.

I am also very grateful to my parents and my sister Kate. Without their exceptional support I would not even have studied anything.

I wish to express my gratitude to Prof Alexei Sourin who helped me to come to Singapore and carefully guided me through the challenges of living in the country which is so different from my home.

The last but not the least, I would like to acknowledge the School of Computer Engineering and Nanyang Technological University for providing me the research scholarship together with excellent research and living facilities.

Maria

Table of Contents

Acknowledgments	i
Table of Contents	ii
Abstract	vii
Candidate's Publications	ix
List of Tables	xi
List of Figures	xiii
List of Abbreviations	xvi
Part I : Preliminary Study	1
1. Introduction	2
1.1 Background	2
1.2 Objectives	4
1.3 Problems and Challenges	7
1.4 Organization of the Thesis	11
2. Literature Review	14
2.1 Gene Expression Regulation	15
2.1.1 Transcription and translation of genes	16
2.1.2 Steroid hormone receptors as regulators of gene transcription	18
2.2 Wet-lab Methods for Gene Expression Regulation Study	21
2.2.1 Identification of transcription factor binding sites	22
2.2.2 Detection of primary target genes	27
2.3 Bioinformatics for Gene Expression Regulation Study	29

Table of Contents

iii

2.3.1 Prediction of transcription factor binding sites	29
2.3.2 Computational models for steroid hormone response elements	34
2.4 High-Performance Computing Technologies for Machine Learning	35
2.4.1 Machine learning	36
2.4.2 Reconfigurable computing	38
Part II : Additive Statistic Model of Hormone Response Element . . .	42
3. Position Frequency Model for Generation of the Consensus HRE Sequence	43
3.1 Understanding HRE Structure through Mutation Analysis	43
3.2 A General Position Frequency Model	46
3.2.1 Construction of the HRE training database	47
3.2.2 HRE homogeneity test with androgen, glucocorticoid, and progesterone	54
3.2.3 Position weight matrix model	56
3.3 Extended Position Frequency Model for HRE Recognition	59
3.3.1 Higher-order PWMs: correlation and justification	60
3.3.2 The first-order weight matrix model	63
3.3.3 The second-order model	65
4. Construction of Markov Model for Description of State Transitions in HRE	67
4.1 A Cascade Markov Model for HRE Recognition	67
4.1.1 Component Markov model	68
4.1.2 Parameter estimation	70
4.1.3 Representation of the HRE complex structure with a cascade Markov model	71

Table of Contents

iv

4.2 The Combined Position-Transition Model and Tiger HRE Finder .	74
4.3 Experimental Results	76
5. Empowering Accurate HRE Modeling with FPGA Implementation . .	79
5.1 Fine-Grained Parallelization with FPGA	80
5.2 Logic Interconnection in the Markov Model	83
5.3 Analysis of Computational Performance and Complexity	89
5.4 Summary and Discussion	90
Part III : Adaptable Model for Flexible HRE Pattern	92
6. Dynamically Adaptable Neural Network for HRE Modeling	93
6.1 A Feed-Forward Neural Network for Sequence-Based HRE Recognition	95
6.1.1 Model definition	96
6.1.2 Network design and training	100
6.1.3 Prediction Results	101
6.2 A Hopfield Neural Network for Structure-Based HRE Classification	103
6.2.1 Symmetric structure of HRE	103
6.2.2 Recurrent neural model	105
6.2.3 Hopfield neural network for modeling the HRE dimer	109
6.3 Global Optimum of HRE Structure Prediction with Dynamic Programming	111
6.3.1 Dynamic programming for alignment of biological sequences	112
6.3.2 The DP-HNN model	113
6.3.3 Dimeric structure prediction for the collection of functional HREs	118

Table of Contents

v

6.4 A Two-Phase Neural Model for HRE Recognition	120
6.5 Results and Discussion	123
7 Achieving Convergence of the Neural Network with Hybrid Computing	128
7.1 Computational Complexity of Neural Network	128
7.2 Dynamically Adaptable Hopfield Neural Network on FPGA	130
7.2.1 Calculation units	131
7.2.2 Neuron and memory units	135
7.2.3 Operation of the DP-HNN	136
7.3 Hybrid Adaptive System for HRE Recognition	139
7.4 Prediction of Progesterone Primary Target Genes	141
7.4.1 The progesterone primary target genes	141
7.4.2 Method for prediction of primary target genes	142
7.4.3 Prediction Results	145
7.5 Summary and Discussion	150
Part IV : Concluding Remarks	153
8. Discussion and Conclusion	154
8.1 Summary on the Thesis Research Contributions	154
8.2 Critical Analysis and Assessment of the Proposed Methods	156
8.2.1 Recognition of HREs using statistic models and neural networks	156
8.2.2 High-performance machine learning based on reconfigurable computing technology	160
8.3 Recommendations for Future Work	161

Table of Contents	vi
<hr/>	
8.3.1 Beyond binding sites	161
8.3.2 Expanding the computational limits	165
Bibliography	167

Abstract

An important step in understanding the conditions which specify gene expression is to recognize gene regulatory areas that are associated with regulation of transcription. Due to high diversity of different types of transcription factors and their DNA binding preferences, it is a challenging problem to establish an accurate model for computational prediction of functioning regulatory elements in promoters of eukaryotic genes. *In silico* modeling of transcription factor binding sites becomes even more complicated in the case of some specialized transcription factors, e.g. nuclear receptors, which interact with their target DNA sequences as homo- or hetero-dimers, thus stipulating the multi-component structure of the binding sites.

This thesis is mostly a methodology research aimed to solve the problem of recognition of symmetrically structured DNA motifs by using bioinformatics tools. Steroid hormone response elements (HREs), which are known to hold partially symmetric structure, are selected for our study.

Addressed in the thesis are two novel methods for recognizing symmetrically structured DNA motifs; their applicability is demonstrated through a set of designed experiments. The first method exploits sequence-specific statistical modeling of the HRE pattern. Though characterized with certain limitations, for the particular problem of accurate HRE recognition, the simple and easily interpretable statistical model largely benefits from application-specific adaptation. The second method is based on a more accurate object-specific motif recognition paradigm, exploiting a two-phase neural architecture. The algorithmic research is preceded by the collection and pre-processing of the HRE training data, thus providing some interesting findings about HRE composition at the early stage of research. The collected dataset of functioning HREs currently has no analogs.

In the statistical HRE modeling, we consider both nucleotide position and inter-nucleotide transition patterns of HRE sequences. We implemented a position-transition recognition model with reference to the complex structure of the HRE motif. A cascade Markov model was invented for modeling the multi-component HRE structure, and it was implemented by hardware using FPGAs (Field-Programmable Gate Arrays). The designed chip was further used as a co-processing unit for computational efficiency when applied for predicting HREs in large genomic

sequences.

Another main contribution of the thesis is the development of a two-phase neural architecture for specific HRE recognition. For higher flexibility of modeling the partially repeated DNA sequences by neural networks, a novel dynamically adaptable recurrent architecture was proposed. That pattern-specific approach outperformed the position-transition model in terms of prediction accuracy, but its prohibitively long computation time made the applicability of such a solution very much limited. A partial hardware acceleration was developed for resolving the computational bottleneck, and a hybrid PC-FPGA recognition tool was further implemented for analysis of hormone primary target genes.

The adaptive multiple-feature pattern recognition framework accelerated with application-specific reconfigurable computing provides an accurate and highly efficient solution for the problem of prediction of nuclear receptor response elements in DNA. The invented approach for modeling DNA motifs reflecting their symmetric structure can also be extended to other regulatory signals with repeated or multi-component structure.

Candidate's Publications

Journal Papers

1. Maria Stepanova, Feng Lin, Valerie Lin (2008) A High-Performance Statistic Model for Recognition of Steroid Hormone Response Elements. *Computational Biology and Chemistry*. Submitted.
2. Maria Stepanova, Feng Lin, Valerie Lin (2007) A Hopfield Neural Classifier and Its FPGA Implementation for Identification of Symmetrically Structured DNA Motifs, *Journal of VLSI Signal Processing Systems*, 48(3): 239-254.
3. Maria Stepanova, Feng Lin, Valerie Lin (2006) In Silico Modeling of Hormone Response Elements, *BMC Bioinformatics*, 7 (4): article S27.
4. Maria Stepanova, Feng Lin, Valerie Lin (2006) Establishing a Statistical Model for Recognition of Hormone Response Elements, *Computational Biology and Chemistry*, 30 (5): 339-347.

Book Chapters

1. Maria Stepanova, Feng Lin, Valerie Lin (2007) A Two-phase ANN Method for Genome-wide Detection of Hormone Response Elements, *Lecture Notes in Bioinformatics*, 4774: 19-29.
2. Maria Stepanova, Feng Lin, Valerie Lin (2006) A Profile HMM for Recognition of Hormone Response Elements, *Lecture Notes in Bioinformatics*, 4146, 15-22.

Conference Papers

1. Feng Lin, Jesslyn Saw, and Maria Stepanova. Methods and Strategies for Construction of a Phylogeny-Adaptive Hormone Response Element Consensus Model. *IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB'08)*, Sun Valley, ID, USA.
2. Maria Stepanova, Lin Feng, Valerie Lin. Design and Development of a FPGA-based Cascade Markov Model for Recognition of Steroid Hormone Response Elements, 7th *IEEE International Workshop on High Performance Computational Biology (HiCOMB 2008)*, Miami Florida, USA.

-
3. Maria Stepanova, Feng Lin, Valerie Lin. A Two-phase Neural Network for Genome-wide HRE Recognition, *2nd IAPR Workshop on Pattern Recognition in Bioinformatics (PRIB 2007)*, Singapore.
 4. Lin Feng, Qi Yutao, Du Zhihua, Maria Stepanova, Cheong Lee Sing, Angela Jean. Computing Architectures & Acceleration for Bioinformatics Algorithms, *GridAsia 2007*, Singapore.
 5. Maria Stepanova, Feng Lin, Valerie Lin. A Cascade Markov Model for Recognition of Hormone Response Elements, *IAPR Workshop on Pattern Recognition in Bioinformatics (PRIB 2006)*, Hong Kong, China.
 6. Maria Stepanova, Lin Feng, Valerie Lin. Tiger HRE Finder – A Tool for Identification of Hormone Receptor binding sites in Query Sequences, *International Multi-Symposiums on Computer and Computational Sciences (IMSCCS 2006)*, Hangzhou, China.

Awards

1. Student Award at the 2nd IAPR Workshop on Pattern Recognition in Bioinformatics (PRIB 2007)

List of Tables

2.1	The most popular methods for detection of protein binding sites in DNA are the DNase I footprinting and the EMSA	27
3.1	Substitution matrix for position-specific alignment of HREs	51
3.2	Results of the homogeneity test for ARE, PRE, and GRE	56
3.3	Results of order justification for the PWM model	62
3.4	HRE prediction accuracy for the first-order PWM model	64
3.5	HRE prediction accuracy for the second-order PWM model	65
4.1	Results of HRE prediction by the five-stage cascade Markov model .	76
6.1	Results of HRE prediction by the FFNN	102
6.2	Results of dimeric structure prediction by the HNN and k-means . .	119
6.3	Accuracy of HRE prediction by the two-phase neural model	122
7.1	Computational performance of the hardware-accelerated system implemented using the Virtex-4 FPGA and the 4-way IBM PC . . .	140
7.2	Primary target score values of human PR-responsive genes	146
7.3	Human genes with the highest scores of being progesterone primary targets	147
7.4	Summary of two different FPGA implementations of the EM-HNN .	152

List of Tables	xii
<hr/>	
8.1	Performance of the proposed methods of HRE recognition 158

List of Figures

2.1	Six levels of gene expression control for eukaryotic genes	16
2.2	Gene expression regulation by steroid hormones	19
2.3	The HRE dimer	20
2.4	Schematic diagram of the EMSA experiment	23
2.5	Using an unlabelled competitor DNAs in the DNA mobility shift assay .	24
2.6	Schematic diagram of the DNase I protection experiment	25
2.7	Detection of hormone primary target genes using translational inhibitors	28
2.8	The frequency-based TFBS modeling paradigm	32
3.1	A query from the Tiger HRE DB system	49
3.2	Aligned GREs sequences	53
3.3	Relative frequencies of the nucleotide T along the HRE sequence for each of the three response element and for the joint dataset of HREs . .	53
3.4	Conservation coefficients for the ARE, PRE, and GRE PFMs, and for the joint PFM	58
3.5	Computing $D_{exp}(\text{CpG}, 1)$ using an example PFM	61
3.6	The ROC curve for HRE prediction by the first-order PWM model . . .	64
3.7	The ROC curve for the second order PWM-based prediction of HREs .	66
4.1	State-transition diagram of a basic Markov model for a DNA sequence .	68

List of Figures

xiv

4.2	The five-stage cascade Markov model for HRE recognition	73
4.3	The combined position-transition model for HRE recognition	75
4.4	Web interface of the Tiger HRE Finder tool	75
4.5	The ROC curves for the combined prediction of HREs in comparison with the first- and second-order PWM models and the five-stage cascade Markov model	77
5.1	FPGA structure for implementation of the cascade Markov model	80
5.2	Design flow for development of the parallel architecture for implementation of the cascade Markov model on FPGA	81
5.3	Schematic diagram of the cascade Markov model implemented on FPGA	82
5.4	Dataflow scheme of the Markov processing element MPE H for the cascade Markov model	84
5.5	The RTL schematic diagram of the TPD unit	84
5.6	The RTL diagram of the MPE H unit	86
5.7	The RTL diagram of the cascade Markov model implemented on FPGA	87
5.8	Data flow during the operation of the cascade Markov model	88
6.1	Feed-forward neural network with four input neurons, five hidden neurons, and one output neuron	96
6.2	The ROC curve for HRE prediction by the FFNN with layout I $\rightarrow$ 40 $\rightarrow$ O	102
6.3	The HRE dimeric structure	104

List of Figures

xv

6.4	Sequence logo representation of the steroid HRE	105
6.5	Convergence of the HNN in a 2D space	107
6.6	An example of recurrent alignment	117
6.7	A two-phase neural model for HRE recognition	121
6.8	Amino acid sequence of the glucocorticoid receptor DBD	126
7.1	A. State flow of the dynamically programmed Hopfield neural system. B. Verification unit controls the HNN recurrence	132
7.2	The matching unit is responsible for the recurrent alignment procedure .	134
7.3	The HNN neuron unit	136
7.4	Configuration and control of the DP-HNN on FPGA	137
7.5	Data flow in the DP-HNN FPGA system	138
7.6	Hardware-accelerated hybrid adaptive system for HRE recognition	139
7.7	Genbank file with human chromosome 22 assembly	143
7.8	Histogram of primary target score values of human PR-responsive genes	146
7.9	HREs in the promoter areas of human <i>MMP1</i> gene and its orthologs . .	148
7.10	Venn diagrams for the hormone primary target genes in vertebrates . . .	149
8.1	HRE prediction ROC curves for the statistic and neural network models	158

List of Abbreviations

DNA – deoxyribonucleic acid
RNA – ribonucleic acid
mRNA – messenger RNA (processed RNA, primer for a protein)
aa – amino acid
bp – base pair (of a DNA molecule)
A – Adenine
C – Cytosine (unless stated otherwise; e.g. a programming language C)
G – Guanine
T – Thymine
(N)HR – (nuclear) hormone receptor
PR – progesterone receptor
AR – androgen receptor
GR – glucocorticoid receptor
ER – estrogen receptor
MR – mineralocorticoid receptor
TR – thyroid hormone receptor
VDR – vitamin D receptor
RAR – retinoic acid receptor
RXR – 9-cis retinoic acid receptor
EcR – ecdysone receptor
DBD – DNA-binding domain (of a hormone receptor)
HRE – hormone response element (a DNA binding site for hormone receptor)
PRE – progesterone response element
ARE – androgen response element
GRE – glucocorticoid response element
ERE – estrogen response element
PCR – polymerase chain reaction
TF – transcription factor
TFBS – transcription factor binding site
NCBI – National Center for Biotechnology Information (USA)
EMBL – The European Molecular Biology Laboratory

List of Abbreviations

xvii

-
- EMSA – electrophoretic mobility shift assay (a wet-lab method of multi-molecular complex identification)
- ChIP – chromatin immunoprecipitation (a wet-lab method of protein-DNA complex detection)
- NMR – nuclear magnetic resonance
- CAT – chloramphenicol acetyltransferase (a bacterial enzyme; the CAT gene is often used in reporter systems to measure the level of a promoter- or tissue-specific gene expression)
- TSS – transcription start site (on DNA)
- CRM – cis-regulatory module
- DB – database
- PFM – position frequency matrix
- PWM – position weight matrix (a frequency-based method of sequence modeling)
- PSSM – position-specific scoring matrix, another name for PWM
- PWM1 – first-order PWM (p. 60)
- PWM2 – second-order PWM (p. 60)
- MM – Markov model
- B – begin (of a sequence for Markov models, p. 68)
- E – end (of a sequence for Markov models, p. 68)
- ML – maximum likelihood (a method of parameter estimation for statistical models)
- ANN – artificial neural network
- FFNN – feed-forward neural network
- HNN – Hopfield neural network
- EM-HNN – exact-match Hopfield neural network (pp. 105, 111, 118)
- DP-HNN – dynamically programmed Hopfield neural network (p. 113)
- ROC curve – receiver operating characteristic curve (a graphical plot of sensitivity vs. specificity balance for a binary classification system)
- AUC – area under (ROC) curve
- RE – random expectation (expected false positive prediction rate)
- PT – primary target (score, p. 145)
- CPU – central processing unit
- HPC – high-performance computing
- ASIC – application-specific integrated circuit
- ASIP – application-specific instruction set processor

List of Abbreviations

xviii

FPGA – field programmable gate array (a hardware acceleration technology based on reconfigurable computing)

CPLD – complex programmable logic device

DSP slice – a programmable logic block in Xilinx FPGA chips

API – application programming interface

HDL – hardware description language

IOB – input/output block

CLB – configurable logic block

RAM – random access memory (memory block of a general-purpose or reconfigurable computer)

PCI – peripheral component interconnect

JTAG – Joint Test Action Group architecture for testing circuit boards using boundary scan)

ALU – arithmetic logic unit

MPE – Markov processing element (p. 83)

MPE F – MPE of a flanking region or a spacer (p. 83)

MPE H – MPE of a half-site (p. 83)

MME – Markov merging element (p. 83)

TPD – transition probability detection unit (p. 83)

PPU – pre-processing unit (p. 133)

RTL – register-transfer level (a way of describing the operation of a circuit in terms of the data transfer)

Part I

Preliminary Study

Chapter 1

Introduction

In this chapter, we introduce the research topic of *in silico* modeling of gene regulatory elements, by the example of hormone response elements, using adaptive computing technologies. In Section 1.1, we briefly describe the biological background of the biomedical problems related to molecular action of steroid hormones. In Section 1.2, we propose and discuss the objectives of the research. In Section 1.3, we address the problems and challenges in various aspects of the project, including development and training of the novel pattern recognition models, and specific adaptation of these models for the case of symmetrically structured DNA motifs. And finally, in Section 1.4, we present the organization of the thesis.

1.1 Background

A large number of ontogenetic and physiological processes within different organisms – from fungi to human – are regulated by a small group of steroid hormones. It can hardly overrate the significance of steroid hormones in the life cycle and development of an individual.

Steroid hormones play a central role in regulation of all aspects of female reproductive activity leading to the establishment and maintenance of pregnancy [1]. They are also essential for male fertility [2]. Some steroid hormones have been implicated in the cardiovascular [3], immune [4], and central nervous function [5], as well as in bone development [6]. It is probably more important that steroid hormones are involved in cancer development [7], and their receptors are used as targets for cancer treatment. In particular, androgens regulate the development and progression of prostate cancer, which is now one of the most frequently diagnosed cancers among elderly males [8]. Growth of uterine leiomyomas, which are the most common gynecological tumors, is also associated with the activity of ovarian steroid hormones [9]. Molecular characterizations of breast tumors revealed important roles for the steroid hormone estrogen in the genesis, progression, and treatment of breast cancers [10].

Molecular effects of steroid hormones are mediated by their receptor proteins.

Steroid hormone receptors are transcription factors which exist in cell cytoplasm in inactive form [11]. Receptors which are activated by binding of hormone molecules are transferred to the cell nuclei and interact with hormone response elements (HREs) in DNA, thus regulating transcription of their target genes. However, gene expression regulation by the steroid hormones is indeed not a straight-forward process.

The entire pathway of cellular effects caused by steroid hormones may include several stages of reaction, and they are yet to be described in details. A multi-stage effect on gene expression appears when hormone primary target genes, that are the genes directly regulated by the activated receptors, produce proteins involved into regulation of other genes, thereby causing the secondary regulatory response. Thus, elucidation of the hormone-mediated regulatory pathway is a challenging problem. Current experimental knowledge about exact mechanisms of regulatory effects induced by steroid hormones is still limited because of the diversity of potential regulatory mechanisms, but at least four mechanisms may be involved into the hormone-induced reaction [12]:

- Classical pathway, when activated receptors bind directly to HREs in promoter regions of their primary target genes;
- Hormone-independent pathways, when hormone receptors are activated by growth factors but still bind directly to HREs;
- HRE-independent pathway, when activated hormone receptors bind indirectly to non-HRE sites through interaction with other transcription factors;
- Cell membrane signaling pathway, which result in fast tissue responses without involving gene expression.

Though the existing experimental techniques, such as Polymerase Chain Reaction (PCR), allow measuring the gene expression level at low cost, they do not elaborate the underlying mechanism of gene expression regulation. In particular, without involving additional expensive and time-consuming experimental technologies, it is essentially difficult of nearly impossible to distinguish between primary target genes of steroid hormone receptors and genes regulated through the induced response. However, for using steroid hormones in hormone ablation treatment of cancer and other diseases, it is highly important to know the exact regulatory

pathway and particularly the direct target genes of hormone receptors; otherwise, with the hormonal regulatory machine being a black box, it is difficult to predict potential side effects and develop methods for specific hormonal action. Recognition of HREs in DNA is thus essential for identification of primary target genes for steroid hormones, and this stage will eventually provide access to understanding the exact mechanisms of hormonal gene expression regulation.

In the past, identification of transcription factor binding sites (TFBSs), and HREs as well, was carried out through experimental means only. Bioinformatics methods have now greatly accelerated the process of recognizing gene regulatory elements in DNA, and the usefulness of *in silico* prediction has increased even more in the post-genomic era when the fully annotated genomic sequences became available. Nevertheless, this is not to say that experimental methods are becoming not important. All the computational methods must be validated through wet-lab experiments using animal models or high throughput experimental technologies. Recognition of HREs by computational methods enables researchers to identify potential targets for further experimental analysis.

1.2 Objectives

Hormone response elements are binding sites for transcription factors belonging to the family of steroid hormone receptors. In general, recognition of TFBSs through experimental research is a slow and tedious task. Through the advent of computational biology, many statistical models and algorithms have been developed to examine regions of the DNA that might harbor TFBSs in a quick and efficient manner, but current computational methods for prediction of TFBSs in DNA are still not as reliable as experimental approaches.

The major challenge for *in silico* recognition of TFBSs is their weak conservation which results in low selectivity of statistical models. In particular, the diverse patterns of the TFBSs make it difficult for naïve statistical approaches to specifically distinguish functional binding sites and neutral DNA. As it has been shown in many experiments [13–15], the true predictions are often accompanied by large numbers of false positives, thus making the results of prediction nearly meaningless. An essential way to improve TFBS prediction accuracy for a particular transcription factor (or a family of those) is to enhance the model selectivity by

employing additional features that are specifically innate to the pattern of interest, though at a cost of losing generality.

Combining two or more independent methods for solving a particular recognition problem is a widely used approach [16]. Its successful applications can be found in machine vision [17], medical imaging systems [18], e-mail filters [19], and many other pattern recognition problems [20]. Typically, the methods being combined exploit different aspects of the pattern so that the probability of detecting an object of interest correctly increases. Similarly, a specifically designed multiple-feature TFBS recognition framework is expected to keep the prediction rate for the underlying TFBS pattern while eliminating the false positive signals, because two or more independent characteristics of a functional site are held by chance with much lower probability. In particular, the feature of dimerization [21] known for response elements of nuclear receptors should serve as an additional characteristic of a functional HRE. Thus, employment of the multi-component HRE structure into a model allows recognition of the dimeric HRE pattern with higher specificity.

The main objective of this thesis is to develop an efficient and robust approach for recognition of the transcription factor binding sites with symmetric or repeated structure. After surveying the literature, we found that this area, while being of great importance for bioinformatics of gene expression regulation, is with plenty of rooms for research. Initially, we intended to study DNA binding sites for progesterone receptor only, as the female steroid hormone progesterone is the primary research interest of the author's co-supervisor, Dr. Valerie Lin from the School of Biological Sciences. But eventually we decided not to lose generality for other symmetrically structured DNA motifs, and considered modeling of progesterone receptor binding preferences as a case study for testing the models.

Three intermediate objectives are defined for developing an accurate solution for the HRE recognition problem and its possible extension to other symmetrically structured DNA motifs.

The first objective is to develop a *statistical model* for the HRE pattern, which should consider HRE as a DNA sequence with reference to its mono- and/or oligo-nucleotide frequency distributions. Statistical modeling is largely a state-of-the-art technology for TFBS recognition, due to its simplicity and generality; and it has already been found to be applicable to a wide range of DNA motifs [22–24], though

models of that type sometimes suffer from low selectivity [25]. Therefore, for the HRE pattern recognition problem, we aim to design a statistical model which, in addition to independent nucleotide distributions, can reflect the specific HRE structural features stipulated by its dimeric and partially repeated composition. Thus, we hope to achieve (possibly significant) increase of HRE prediction specificity while keeping the advantages of the statistical models – simplicity, sensitivity, rapidity, and interpretability.

The second objective is to explore an adaptive pattern recognition paradigm for more accurate recognition of the HRE motif using neural networks. The *neural model* considers the HRE sequence as an integral object, without splitting it into independent oligo-nucleotide components, and thus provides an entirely different point of view to the pattern of interest. The intrinsic limitations of the frequency-based approaches has been known since recently [26], while the modern neural networks and kernel-based approaches, when properly trained, allow modeling integral objects in multi-dimensional space with high precision.

Therefore, for the problem of HRE modeling, we aim to design a neural network model, which, when trained with an extensive set of functional HREs and opposed with neural DNA sequences, should be able to distinguish between them with high accuracy. For higher specificity, the approach should also employ the underlying knowledge about the dimeric partially repeated HRE structure. Additionally, the designed neural network should overcome its rigid structure for modeling biological sequences so as to reflect that they are subject to short mutations including insertions and deletions. For statistical models, this problem is solved by a preceding sequence alignment, and for the neural network design, the model of self-organizing network is usually used. In order to avoid the data-demanding self-organizing networks, we propose a novel adaptation approach based on dynamic programming, for simpler and highly flexible modeling of the HRE pattern.

Finally, the third objective is to solve the complex computational problems with affordable computing resource and reasonable execution time. To achieve the best prediction accuracy when dealing with long genomic sequences, we decide to develop a *hardware-acceleration* solution for the proposed HRE recognition models. The benefit from using such advanced computing technologies is becoming obvious as the amount of available biological data keeps growing exponentially. In particular, with the continuously increasing number of fully sequenced genomes being reported each

year, it has already become necessary to involve high-performance computing solutions and acceleration techniques for the complex and flexible approaches to DNA processing to be efficient in practice [27, 28].

The problem of recognizing HREs in large genomic sequences requires fast processing of each single operation rather than distribution of resources and computations among different processors or computers; that is why reconfigurable computing is preferred to conventional cluster-based grid computing. To date, the field programmable gate array (FPGA) technology of application-specific hardware design provides the best combination of development costs and computational performance.

The application-specific on-chip systems have already been widely used for the problems of digital video encoding [29], computer interconnections [30], cryptography and telecommunications [31], automotive and space-flight applications [32]. Eventually, a more general objective of the research project is to study the applicability of the hardware acceleration technologies by the example of FPGA to the problems of pattern recognition in bioinformatics, which is now a promising area.

1.3 Problems and Challenges

Briefly, we intend to develop computational methods for efficient *in silico* recognition of symmetrically structured DNA motifs. We should also design and implement high-performance hardware architectures for the proposed models so that they are effective in real applications. The accuracy and throughput of the hybrid CPU-FPGA solutions is going to be tested using the case of steroid HREs. The computational models, when trained using a collection of experimentally validated HREs, should be able to predict binding sites for steroid hormone receptors with reliable sensitivity and, what is especially important, with high specificity. The problems that are addressed in this thesis can be summarized as follows:

1) *Selection of HREs for training.* For investigation into the structure of the HRE motif and construction of a reliable training set for the computational models, it is necessary to extensively search for experimentally validated response elements. As a primer to the research, we initially scanned public databases that house collections of annotated TFBSs. For the case of steroid receptors we have found only approximately 50 response elements with reliable annotation. Although most of the statistical models,

due to their intrinsic simplicity, could be trained with such a small amount of data, they would return higher variance, and hence lower reliability, for the resulting outputs. On the contrary, adaptive models, including multiple-feature frameworks and neural networks with numerous tuning parameters, require large amount of data for proper training; otherwise, they tend to become over-fitted and fail to return reliable results when tested with previously unseen data. That is, a representative collection of functional HREs, meaning a large but non-redundant one, should be generated. Additionally, using this collection, we should be able to validate previously reported homogeneity of the family of HREs. For training of the statistical models, it is also necessary to propose an alignment scheme in order to consider the specific behavior of the HRE nucleotide composition which varies along the HRE sequence.

2) A *statistic position-transition model* should be designed and trained for recognition of HRE motifs in DNA. Exploiting nucleotide transition pattern represented by the Markov model in combination with a conventional position-specific nucleotide distribution modeled by a position weight matrix (PWM), we hope to enhance the specificity of HRE recognition and increase overall prediction accuracy. Additionally, since it is known that nucleotides of most of TFBSs exert interdependent effects on the binding affinities [26], we also have to consider higher-order position models, including di- and tri-nucleotide HRE composition. However, first, it is necessary to confirm that higher-order position and transition models can indeed provide us with meaningful information that cannot be inferred from the mono-nucleotide HRE composition.

The multi-component and partially repeated HRE structure which includes two half-sites accompanied by flanking regions, each with its own impact to protein-DNA interaction, should also be reflected by the statistic model. A challenge is that the PWM model allows little flexibility in application-specific design; that is, for the purpose of modeling the dimeric HRE structure, an extension of the Markov model's architecture should be explored. We suppose that multiple component models are able to represent the complex HRE structure precisely if each of these models is properly trained together with conditional transitions between the component models. We should then test the ability of the position and transition models to improve each other's performance when used in combination, in order to make a conclusion whether the combined prediction scheme allows us to get rid of significant amount of false

positives in comparison with standalone position and transition models.

3) For object-specific recognition of symmetrically structured HRE motifs, robust and flexible pattern recognition frameworks such as those which exploit *neural networks* should be developed. The main problem is to correctly recognize HRE sequences while reflecting the HRE dimeric structure for higher selectivity. The assumption under the design requirement is inferred from molecular biology which states that even if a sequence is more or less similar to the HRE consensus but does not hold a repeated structure, then it is unlikely involved in the dimeric protein-DNA interaction. Thus, it is necessary to design a two-phase neural architecture operating as an ensemble of two neural models for preliminary recognition of putative motifs based on their nucleotide composition and for further accurate prediction of HRE dimeric structure. While the former task can be accomplished by a feed-forward neural network, the latter will require involvement of a recurrent neural network which should be specifically designed and trained for recognizing HRE dimeric structure. The applicability of recurrent neural networks to the problems of motif modeling has recently been reviewed [33].

Though the feed-forward neural network itself provides a very precise approximation of the training set, its specificity, when measured using an independent testing set, may be not so high [34]. Involvement of the two-phase prediction framework for reflecting the sequence and structure composition of the HRE pattern is expected to allow increasing the specificity of HRE motif prediction while keeping the prediction sensitivity close to that returned by the feed-forward neural network. However, when we deal with adaptive multiple-feature frameworks, proper learning will be a key point. The challenging process of neural learning should be maintained by balancing its training set for robust training, design of the data-specific neural architecture, and proper selection of learning parameters. If not designed properly, the highly adaptive neural system, even when characterized by a good learning accuracy, may fail to operate with previously unseen data.

4) *Dynamic programming for enhancement of HRE dimeric structure recognition.* The problem that should be considered when designing the two-phase neural system for HRE recognition is that inflexible architecture of the neural networks may cause biologically unreliable results of HRE structure prediction.

Indeed, the dimeric HRE is composed of two half-sites that interact with two

zinc-fingers of a receptor's DNA-binding domain and are separated by a short region of non-interacting DNA. This non-interacting region is approximately 3bp-long, as it has been confirmed by numerous experiments (reviewed in Section 3.1), so that the contact points of the HRE half-sites are separated by exactly a helical repeat of DNA, but short mutations, such as insertions and deletions, can also be tolerated. However, due to inflexibility of the conventional structures of neural networks, these mutations may cause oscillations of HRE structure prediction due to frame-shifts caused by insertions and deletions, and thus increase the number of false negative outputs and provide biologically meaningless results. Thus, for higher sensitivity of HRE prediction, we are to design an extension for the neural system to be dynamically adaptable. The adaptability of the neural networks would allow us to model HRE dimeric structure with higher flexibility and robustness, since typical short mutations of biological sequences will also be incorporated. We hope that the extension would also be applicable to the cases of other partially symmetric or repeated motifs, which is currently of great importance for the problems of bioinformatics of gene expression regulation.

5) High-performance *FPGA architectures for hardware acceleration* of the computational models should be designed and implemented for achieving high accuracy in tolerated execution time. The multiple-feature frameworks for HRE recognition can cause prohibitively long computation time when applied to real genomic sequences due to excessive calculations. The high-performance computing technology based on reconfigurable computing should be explored in order to make the algorithms effective for high-throughput analysis.

Simple statistic methods like PWMs can take advantage of massive data parallelism when each processor works independently on its portion of the input sequence. The resulting processes do not require much communication between one another due to straight-forward data processing scheme, whereas, often, low accuracy of TFBS prediction is achieved [35]. More sophisticated methods exploiting multiple-feature prediction frameworks may be useful for establishing more accurate models. However, when dealing with gigabytes of fast growing genomic data, computational speed will eventually determine the applicability of the desired algorithms which compromise the best precision of computational models.

For recognition of HREs by the two-phase neural architecture, operation of the

recurrent neural network designed for predicting HRE dimeric structure will be the bottleneck, because it requires numerous iterative computations not only for learning, but for each input sequence. Since the states of the recurrent neural model depend on the previously processed data due to model feed-back, there is little possibility for cluster-based acceleration by massive data parallelism. Some parallelism may be achieved when neurons, or groups of those, are mapped to different processor for independent computations, but such an implementation will incur extremely high communication overheads due to frequent library function calls for transmitting neuron outputs between the processors. However, the problem of computation delays due to over-communication and bus contentions can be avoided by using the FPGA technology which, in addition to the on-chip parallelism, provides application-specific logic interconnections for data flow. In contrast to that of distributed computing technologies, the communication cost within an FPGA chip will be very low.

The time-consuming stages of the multiple-feature HRE recognition frameworks should be described at the low level using hardware description language for further mapping to semiconductor hardware circuits. Thus, the on-chip implementations of the adaptable neural network and the multi-stage Markov model will be developed and used as co-processing units for HRE prediction by the two-phase neural model and the position-transition model respectively. Having publicly available genomic sequences of vertebrate species in hand, we are to test the performance of the CPU-FPGA hybrid systems and compare them with high-performance software solutions. The on-chip versions of the neural network and the multiple-component Markov model are expected to provide access to the results which are currently inaccessible using general purpose CPUs.

1.4 Organization of the Thesis

The thesis is organized so as to describe the proposed models for recognition of symmetrically structured DNA motifs in a logical flow. It is divided into four parts.

The first part is a primer to the research and contains two chapters. In this Chapter 1, we describe the background of the research domain, outline the objectives of the research work, and introduce the challenging problems. Chapter 2 contains literature review which covers the molecular biology areas related to the thesis topic, and provides a survey of existing methods for *in silico* investigation into gene

expression regulation. In particular, we explain how the steroid hormones are involved into the process of transcription regulation and describe the medico-biological aspects of their activity. The survey on bioinformatics approaches covers the methods for TFBS recognition in general, and also the approaches specifically developed for the symmetrically structured motifs. In the same chapter, we also conduct a study on machine learning and high performance computing technologies which will be further extensively developed for bioinformatics and biomedical problems.

The second part describes the additive statistic model of the HRE pattern. It presents our research contributions at the first stage of *in silico* HRE modeling. Namely, we propose the position-transition HRE model which is further enhanced with hardware acceleration technology. Chapter 3 describes the position frequency-based model of HREs implemented using the weight matrix method. The first-order weight matrix, which is a known solution for modeling of biological sequences, is extended to the second-order model. In that Chapter, we also describe the collected dataset of HREs which is further used for training and testing of the weight matrix model and other computational models. In Chapter 4, we propose the profile Markov model architecture for recognizing the HRE motifs exploiting their inter-nucleotide transition pattern. A multi-stage architecture for the Markov model is introduced with reference to the known multi-component HRE structure. Eventually, as the Markov model and the weight matrix model represent different approaches for modeling the nucleotide sequences, we also test their ability to work in cooperation for recognition of HRE motifs. Additionally, in Chapter 4, we describe the publicly available tool Tiger HRE Finder where the two statistic models operate online. In Chapter 5, we consider the computational complexity of the proposed position-transition model for HRE recognition, and introduce an FPGA-based hardware acceleration of the multi-stage Markov model. In that chapter, we also summarize and discuss methods and results described in Part II.

The third part of the thesis presents a CPU-FPGA hybrid neural architecture for more accurate and sensitive HRE prediction. In Chapter 6, we describe the two-phase neural model which considers the repeated structure of the HRE motif. First, we introduce the feed-forward neural network for HRE recognition without reflecting its dimeric structure. Then, for dimeric structure prediction, we propose a recurrent neural network solution. The relevant biological background with investigation into the idea

of the HRE dimeric structure is also given. Both exact-match and dynamically adaptable recurrent neural models for HRE structure prediction are introduced. The feed-forward and recurrent neural models are further used for establishing a multiple-feature prediction framework, and the results of HRE prediction by the two-phase neural model are described. In Chapter 7, we develop the hybrid neural architecture with partial FPGA acceleration for accurate and flexible HRE recognition. First, we provide the estimation of computational complexity for the two-phase model, which motivates the use of high-performance computing technology. Then, the on-chip implementation of the dynamically adaptable recurrent neural network is described. Based on this implementation, we complete the hybrid neural machine for precise recognition of HREs on genomic scales. Finally, we test the applicability of the proposed HRE recognition algorithm to the problem of analysis of hormone primary target genes.

In the last part of concluding remarks, we provide summary and conclusions to the described research. In Chapter 8, we also give an outlook for future research, both for bioinformaticians who may want to develop more sophisticated algorithms for better prediction of functional regulatory elements on DNA, and for hardware engineers, who are expected to contribute into a better time/accuracy balance for the high-performance solutions in bioinformatics.

Chapter 2

Literature Review

Completion of human genome sequencing and accelerating pace of sequencing the genomes of many other organisms has undoubtedly changed the outlook of biological research. Life scientists and bioinformatics researchers define the present status of research as "the post-genomic era" which marks the need to approach the study of living organisms with a different perspective. To understand this transition, we must recall that much of the research in biology and medicine in the past two to three decades has been characterized by the efforts to identify the gene(s) responsible for a given biological function (or disease, for much of human pathology), isolate it and determine the DNA sequence bearing the information for it. Today, for many living organisms, complete and well annotated DNA sequences are available, and this list is expanding continuously; that is, the researchers already have all the necessary DNA information on hand. In addition to providing experimentalists with previously unavailable information, the post-genomic era accumulated huge amounts of DNA data which lacked attention in the past due to limited knowledge and resources and complexity of underlying biological processes. It has led to dramatic and rapid development of an entirely new field of study known as *bioinformatics* which is sometimes referred to as computational biology.

Though undefined precisely, the bioinformatics discipline could be described as a hybrid of mathematics, life science, and computer science. Generally speaking, molecular biology, genetics, biochemistry and biophysics provide basic knowledge for design of databases and mathematical models which are then implemented as algorithms for simulation of the biological processes. Thus, "*in silico* biology" gives us a better understanding of relevant nature principles since the best way to understand something is to reconstruct it, while the low cost of computational experiments provides access to wider fields of research. While the previous decades could mostly be referred to as an accumulation period, resulted in enormous amount of biological data being stored in different databases (namely NCBI, EMBL, Swiss-Prot, etc.), in the past years, we can see more effort being made to analyze these data in order to extract biological knowledge and use this knowledge to gain new information.

“Accumulation era” is conventionally thought to be officially opened in 1980 when the first special issue of Nucleic Acids Research journal on bioinformatics was published; though, the first databases PIR [36] and PDB [37] with protein sequences appeared several years earlier. Since then, the notable events in the biological database development were: sequencing of the complete DNA of phage lambda in 1982 [38], sequenced DNA of viruses in the mid-80s, and the first complete genome of a bacteria *Haemophilus influenzae* in 1995 [39]. By the time when the Human Genome Project was established in the late 1990s, several sequence alignment [40] and database search [41] algorithms had already been reported.

The modern area of *computational genomics* underwent rapid acceleration after the announcement of fully sequenced human genome in 2001 [42, 43], and now we have dozens of fully sequenced mammalian genomes, as well as numerous vertebrates, viruses and bacteria. Since the early 2000s, the necessity for development of algorithms for extracting knowledge from this fortune of data has been evident.

In this chapter we provide a review of literature related to the thesis topic. In Section 2.1, we give a brief introduction to the idea of gene expression regulation. In Sections 2.2 and 2.3, we provide a survey of existing laboratory and computational techniques for investigation of TFBSs, respectively. In Section 2.4, we survey the field of machine learning and the idea of high-performance solutions for machine learning in bioinformatics.

2.1 Gene Expression Regulation

Gene expression is a fundamental biological event within any living organism. Gene expression is the process by which the inheritable information in a gene is made into a functional gene product, such as protein or RNA. In other words, in addition to carrying the hereditary information, genes are the blueprint for the synthesis of proteins that make up a cell, and the Central Dogma of molecular biology is a framework for understanding of the transfer of sequence information between sequential information-carrying biopolymers DNA, RNA and proteins. In particular, the DNA biopolymer is transcribed into RNA through the action of DNA polymerase II, and then the resulting mRNA is translated into a protein via ribosomes in the cytoplasm [44]. Every stage of expression process is thoroughly regulated for each gene in each cell, so that the resulting living organism lives, as a matter of fact.

2.1.1 Transcription and translation of genes

Numerous cell types in a multicellular organism differ dramatically in both structure and function. For this reason, and because cell differentiation is often irreversible, biologists originally suspected that genes might be selectively lost when cells differentiated. We now know, however, that cell differentiation is mostly associated with variations in gene expression rather than with any changes in nucleotide sequences of the genomes.

There are several steps in the “DNA to protein” pathway, and each of them can theoretically be regulated. Thus a cell can control the proteins it makes by (1) controlling when and how often a given gene is transcribed (transcriptional control), (2) controlling how the RNA transcript is spliced or otherwise processed (RNA processing control), (3) selecting which completed mRNAs in the cell nucleus are exported to the cytosol and determining where in the cytosol they are localized (RNA transport and localization control), (4) selecting which mRNAs in the cytoplasm are translated by ribosomes (translational control), (5) selectively destabilizing certain mRNA molecules in the cytoplasm (mRNA degradation control), or (6) selectively activating, inactivating, degrading, or compartmentalizing specific protein molecules after they have been made (protein activity control) (see Fig. 2.1).

Fig. 2.1 Six levels of gene expression control for eukaryotic genes.

For most of the eukaryotic genes, transcriptional control is paramount. This makes sense because, of all the possible control points illustrated in Fig. 2.1, only transcriptional control ensures that the cell will not synthesize superfluous intermediates. An enzyme RNA-polymerase (I, II or III) performs the transcription process and a transcribed region is used as a blueprint for the resulting RNA molecule. However, the RNA polymerase is hardly able to initiate transcription on its own. Genetic analyses in bacteria carried out in the 1950s provided the first evidence for the existence of gene regulatory proteins that turn specific sets of genes on or off [45]. It is, in fact, the *transcription factor* proteins which regulate the ability of RNA-polymerase to initiate transcription [46].

Transcription factors are DNA-binding proteins which compose complex units of proteins interacting with RNA-polymerase and each other [47]. Transcription factors charged with transcription regulation bind to DNA sequences in gene regulatory areas which include [44]: *i.* the basal (core-) promoter to which TATA-binding protein and its auxiliary proteins bind; the core-promoter is located near the transcription start site (TSS) and is similar for all protein-coding genes; *ii.* the upstream promoter to which transcription factors bind; this region highly differs from gene to gene; *iii.* the enhancers and silencers, which are located far from transcription start sites and are charged with fine regulation. Transcription factor binding sites, or response elements, are usually short DNA regions (5-25 base pairs long) with high affinity to DNA binding domains of transcription factors.

One might have predicted that the information present in genomes would be arranged in an orderly fashion, resembling a dictionary. Although the genomes of some bacteria seem quite well organized, the genomes of most multicellular organisms are surprisingly disorderly, and small bits of protein-coding DNA are interspersed with large blocks of seemingly meaningless DNA. Some sections of genomes contain many genes, and others, possibly much longer ones, lack genes. Proteins that work closely with one another in the cell often have their genes located on different chromosomes, and adjacent genes typically encode proteins that have little to do with each other in the cell. Decoding genomes is therefore far from being simple. Even with the aid of powerful algorithms and large databases including fully sequenced genomes of different organisms, it is still difficult for researchers to locate definitively the beginnings and ends of genes in the DNA sequences of complex genomes, much less

to predict when each gene is expressed during the life of the organism. Although the DNA sequence of the human genome is now known, it have already taken years and will probably take years more for scientists to identify every gene with its relevant expression control mechanisms.

2.1.2 Steroid hormone receptors as regulators of gene transcription

The necessity of using hormonal therapy for treatment of different diseases, together with existing conception about potential cancer risks associated with such treatment [48–51], gave rise to numerous attempts to understand the molecular mechanisms of steroid hormones' action. It has been established by animal experimentation that modification of the hormonal environment by surgical removal of endocrine glands, by pregnancy, or by exogenous administration of steroids can increase or decrease the spontaneous occurrence of tumors or the induction of tumors by applied carcinogenic agents [52]. The incidence of tumors in humans could also be altered by exposure to various exogenous hormones, singly or in combination. In spite of numerous data obtained during these studies, the entire pathway of different cellular processes regulated by steroid hormones is still unclear.

Steroid hormone molecules have relatively small molecular mass (~300). They are synthesized of cholesterol and secreted by endocrine cells [53] which can belong to different organs and tissues, including ovaries and placenta which express progesterone, adrenal cortex which expresses androgens and glucocorticoids, and even adipose tissue which may express estrogens [54]. Then, the hormone molecules are transported in the bloodstream or other extracellular fluids (usually bound to carrier proteins). Being hydrophobic, they are able to pass through the plasma membrane by simple or facilitated diffusion. Inside a target cell, steroid hormones of each type strongly, but reversibly, bind to specific receptor proteins. Unlike most water-soluble hormones which are removed and/or broken down within minutes from entering the blood, the steroid hormones persist in the blood for hours, and the thyroid hormones are there for days.

The steroid hormone receptors are intracellular transcription factors that exist in inactive apoprotein forms either in the cytoplasm or nucleus [12]. Binding of a hormone results in an allosteric change of conformation of the receptor (this process is known as “activation of a receptor”) that raises the affinity of the receptor to DNA; it

allows the receptor to bind to specific parts of DNA molecule inside a nucleus and to adjust transcription of the *cis*-linked genes (see Fig. 2.2). In addition to regulating transcription, steroid hormones occasionally regulate gene expression by affecting mRNA stability and translational efficiency [53].

Fig. 2.2 Gene expression regulation by steroid hormones.

The superfamily of steroid hormone receptors is encoded by a large number of genes, and represents the largest known family of transcription factors in eukaryotes [55]. It includes receptors of the steroid hormones estrogen (ER), progesterone (PR), glucocorticoid (GR), mineralocorticoid (MR), and androgen (AR). In addition, it includes receptors for thyroid hormone (TR), vitamin D (VDR), retinoic acid (RAR), 9-*cis* retinoic acid (RXR) which heterodimerizes with a peroxisome proliferators-activated nuclear receptor (PPAR) [56] and liver X receptor (LXR) [57], and ecdysone (EcR). More than 30 genes encoding at least 75 protein molecules (note that hormone receptors can have different isoforms with different DNA specificity, regulation preferences, or hormone affinity) have already been identified, and new members of the superfamily are being reported frequently, such as the receptors which respond to dioxin. Using new biotechnologies, molecular biologists and biochemists have identified protein receptors for which the ligands have not yet been identified, thus giving birth to a class of "orphan receptors".

Like many eukaryotic transcription factors, nuclear hormone receptors are composed of domains that correspond to discrete functions [53, 58] (and references therein). By comparison of amino acid sequences of receptor proteins, it was demonstrated that the amino termini of nuclear receptors vary considerably in length and composition, but the sequences of the DNA- and ligand-binding domains are well conserved among the superfamilies. Moreover, DNA-binding specificity of the receptors was found to be encoded solely by their DNA-binding domains and could be switched by swapping the corresponding segments between different receptors [59].

The true nature of the receptors for progesterone, androgen, estrogen, glucocorticoid and mineralocorticoid hormones together with the differences between hormone receptors has been elucidated very recently. Much work is in process, and the area is currently one of primary research interest to many biotechnology companies and research groups. We believe that in the next twenty-thirty years, some of the most profound advances in drug therapy will come out of the concepts first observed with steroid hormone receptors.

Hormone Response Elements

Many transcription factors, including the steroid hormone receptors, bind DNA as dimers [60]. Dimeric binding means that two zinc-fingers of the protein's DNA-binding domain interact each with its own short DNA region, and these two regions are thus separated by a short spacer. DNA sequences responsive to steroid hormone receptors have been termed as hormone response elements (HREs). Hormone receptors activate expression of specific genes by binding to HREs.

Fig. 2.3 The HRE dimer.

During evolution, steroid receptors (AR, PR, GR, MR, and ER) likely diverged in a coordinate manner with their DNA-binding targets to provide non-overlapping functions [61]. Most dramatically, ER evolved the amino acids in the DNA recognizing α -helix that provided new discriminating contacts to the central nucleotide of the half-site sequence [62]. However, due to the confinement of the highly conserved structural features of the DNA-binding domains, functional integrity, and

the relatively recent common ancestry [59], it is possible that steroid receptors with an identical DNA recognition α -helix (AR, GR, PR, MR) are capable of binding both to a common variety of sequence, as well as divergent sequences that display more receptor selectivity. Eventually, several classes of HRE have been characterized. The ARE/PRE/GRE class which consists of response elements for androgen, progesterone, and glucocorticoid receptors and has a consensus sequence AGTACAnnnTGTTCT (Fig. 2.3) has been studied the most extensively. Mineralocorticoid receptors can also act via this response element. The estrogen response element (ERE) consensus sequence GGTCAnnnTGACC can be converted to a functional ARE/PRE/GRE by just changing two bases. The sequences are imperfect repeats, either direct or palindromic, to which the receptor dimers bind [63]. Conventional length of the HRE spacer between the two half-sites is 3 bp, and any change of it leads to certain decrease of response element's functionality [64].

In conclusion to this section, gene expression is regulated in several ways, and the primary way is transcription regulation which can be studied at the DNA and mRNA level. Steroid hormones are a group of ligands that mediate the regulation of transcription by their nuclear receptors. Hormone receptors recognize response elements in DNA and induce or suppress the transcription of their target genes. The steroid hormones androgen, progesterone, and glucocorticoid have been reported to share similar response elements, and their response elements have also been proved to carry distinct features. The first important feature is the structure with two repeated half-sites separated by a 3bp-long spacer. The second is that the right half-site is better conserved than the left one (reviewed in Section 3.1). The third and the most important is the observation that there exist single nucleotide positions within each of half-sites which play important roles of contact points to the bound receptor protein, and therefore are more conserved than others. Knowledge of the HRE structure is a fundamental for reconstruction of a model for highly specific HRE recognition.

2.2 Wet-lab Methods for Gene Expression Regulation Study

The sequencing of complete genomes together with experimental investigation into the gene expression control mechanisms has revealed several fundamental principles of gene expression regulation. In particular, expression level of almost all the vertebrate genes is subtly regulated by at least one, but often multiple, transcription

factors (TFs). TF proteins bind to specific short regions of the DNA molecule (transcription factor binding sites, or TFBSs) within regulatory areas of their target genes. The functional availability and ordered combination of TFBSs in the regulatory areas form promoter modules that are associated with the control of gene expression. Studies of TFBSs can provide essential information for reconstruction of regulatory networks and the functional context of specific genes.

In vitro analysis is the only method for accurate detection of protein-DNA complexes, but it is characterized by certain challenges due to the fact that TFs usually bind to multiple target sequences with varying affinity, and they also often regulate expression of multiple genes. *In silico* analysis still cannot provide comparably accurate results, but presents a versatile extension to current *in vitro* methods. The main problems for computational analysis are that TFBSs are often located in non-coding DNA, degenerate in their sequence, and are relatively short (5–15 nucleotides). Searching for such low-information content objects within large amounts of genomic DNA using computational methods typically results in a large number of randomly occurring false positive predictions. That is, the necessity of experimental validation of computational predictions is still very important.

Additionally, as the experimental knowledge is usually used for development and training of computational models, it is necessary for a bioinformatician to know precisely what information is indeed provided by any molecular biology method, which assumptions are made for that, and, therefore, how reliable is the reported finding. For this purpose, a brief insight into experimental techniques could be useful.

2.2.1 Identification of transcription factor binding sites

There exist several methods for investigation to protein-DNA interaction. It is important to note in advance, though, that one should distinguish methods which allow detecting protein-DNA complexes directly, and indirect methods of protein binding site detection applied when a putative binding site is inserted into a promoter region of a reporter gene and then the difference in expression level of this gene is measured in the absence and presence of the regulatory protein. Though the inference about protein-DNA binding based on the observation of the corresponding expression level is usually reliable, it may also lead to bad errors because the real pathway of gene expression regulation is not elaborated [65].

With certain exceptions, most of the methods for protein-DNA binding detection

can be subdivided into several major groups according to the idea they imply for detection of the protein-DNA complex.

A. Shift assays are based on the assumption that DNA with bound protein is heavier than empty DNA. *Electrophoretic mobility shift assay (EMSA)* [66] provides a powerful tool for detecting protein binding to specific DNA sequences. The method assumes binding of the protein to a radio-labeled DNA fragment (probe) *in vitro* which is followed by electrophoretic separation of protein-DNA complexes from unbound DNA on non-denaturing polyacrylamide gel. Eventually, the experimentalist will see a retarded band on the gel picture because the DNA bound to the protein moves more slowly through the electrophoresis than a free DNA.

One or more proteins binding to the DNA fragment may be identified thereby. Generally speaking, the larger the DNA-protein complex, the greater is the extent of retardation within the gel [67]. The principle of the DNA mobility shift assay is illustrated in Fig. 2.4.

Fig. 2.4 Schematic diagram of the EMSA experiment. Binding of a cellular protein (B) to the radioactively labeled DNA causes it to move more slowly upon gel electrophoresis and hence results in the appearance of a retarded band upon autoradiography.

Fig. 2.5 Using an unlabelled competitor DNAs in the DNA mobility shift assay. If the unlabelled competitor is capable of binding the same protein, it will do so, and the retarded band won't be observed.

The DNA-related specificity of the protein within the complex can be verified by competing for binding with non-radio-labeled DNA fragments. If a molar excess of a DNA fragment capable of binding the same protein is introduced into the binding reaction, much of the protein will bind to the unlabeled DNA leaving less protein available for binding to the probe. This will lead to a reduction or elimination of the band corresponding to the complex formed by that protein (see Fig. 2.5).

EMSA is one of the most sensitive methods for studying the binding properties of the protein for its site. It can be used to deduce the binding parameters and relative affinities of the protein for one or more sites, or for comparing the affinities of different proteins for the same sites. It can also be employed to study higher-order

complexes containing multiple proteins. For example, a single protein binding to a single site would generate one predominant shifted complex. If another protein is bound on top of that, it will generate an additional shift or “super-shift” on the autoradiography. Finally, EMSA can be used to study protein-specific or sequence-dependent DNA binding.

B. Footprinting methods use the idea that nucleotide bases in the DNA region where the protein binds show altered reactivity towards a chosen DNA modifying agent. The most popular approach exploits the *DNase I protection* method.

Fig. 2.6 Schematic diagram of the DNase I protection experiment. DNase I attacks and cleaves DNA by binding to the minor groove of the double helix. Location of the region of DNase I protection relative to a marker can be determined by electrophoresis and size determination of digestion products resulting from mild DNase I digestion.

Because of its simplicity, DNase I footprinting is widely used for both identifying and characterizing DNA-protein interactions. The concept is that a partial digestion by DNase I of a uniquely radioactively-labeled fragment will generate a ladder of fragments, whose mobility on a denaturing acrylamide gel and whose positions in a subsequent autoradiograph will represent the distance from the end label to the points of cleavage. The bound protein prevents binding of DNase I in and around its binding site and thus generates a “footprint” in the cleavage ladder. The distance from the end label to the edges of the footprint represents the position of the

protein-bound site in the DNA fragment. The exact position of the site can be determined by electrophoresis of a DNA sequencing ladder alongside the footprint. Illustration of the DNase I protection method is given in Fig. 2.6.

However, DNase I cannot bind directly next to the DNA-bound protein because of the steric hindrance. Hence the footprint gives a broad indication of the binding site, generally 8–10 bp larger than the site itself.

Other reagents could also be used instead of DNase I, namely, dimethylsulfate for methylation assay, exonuclease III, ethylnitrosourea for phosphate ethylation assay, KMnO_4 , Diethylpyrocarbonate, OsO_4 , Fe(II) EDTA for hydroxyl radical footprint, psoralens, phenantroline/Cu [68], and others. One major disadvantage of all genomic footprinting procedures is that many target molecules are required to obtain a complete sequence profile, because chemical cleavage reactions are involved.

C. Chromatin immunoprecipitation (ChIP) assay might be used for genome-wide determination of protein binding sites [69]. The fundamental of this assay is that DNA-bound proteins can be cross-linked to the part of chromatin where they are located. After the chromatin with immobilized proteins on it is fragmented, the protein-DNA complexes are immunoprecipitated using an antibody specific for the protein. The DNA from the isolated protein/DNA fraction is then purified and identified through PCR. Besides, a technique that combines ChIP assays with microarray technology has recently been developed (so called “ChIP-on-chip”) [70]. The main disadvantage of ChIP assays is the requirement for highly specific antibodies for the proteins which are usually expensive, thus delaying the spread of this highly accurate method.

Finally, there are methods for direct visualization of protein-DNA complexes, namely, the *X-ray crystallography* [71] and *NMR spectroscopy* [72] which provide a source for investigation into the structural details of protein-DNA interactions at the molecular and/or atomic level. Though very informative, these experiments are very time-consuming.

The most popular methods have been the DNase I footprinting and the EMSA [73]. Their advantages and disadvantages are summarized in Table 2.1. In short, the main advantage of the DNase I footprinting is that it allows using long probes and detecting nearly exact locations of binding sites for several proteins (though sometimes approximately, especially in the case of close or shared sites or cooperative binding).

The EMSA method requires additional procedures to precisely locate binding sites in a long DNA probe, but allows accurate distinguishing among protein complexes.

Table 2.1 The most popular methods for detection of protein binding sites in DNA are the DNase I footprinting and the EMSA.

DNase I footprinting	EMSA
<ul style="list-style-type: none"> • Reveals approximate locations of binding sites on the probe • Reveals locations of binding sites for multiple proteins on the same probe (but not for multiple proteins which bind the same site) • Large footprint can suggest that two or more proteins bind to adjacent elements on the probe 	<ul style="list-style-type: none"> • Location of binding sites can be determined only by mutagenesis of the probe or methylation interference • Can reveal a unique complex for each protein that binds the probe even if multiple proteins bind to the same element • Can be difficult to determine if a complex contains two proteins that cooperatively bind to adjacent sites
<p><u>DNase I footprinting</u> is particularly useful for scanning large DNA fragments (50-200 bp)</p>	<p><u>Electrophoretic Mobility Shift Assay</u> is useful for investigation into protein-DNA interactions on a small DNA fragment and for determining cell-specific binding activity</p>

2.2.2 Detection of primary target genes

After the invention of the microarray technique and the polymerase chain reaction at the end of the previous century, the problem of regulated gene expression have been investigated in details for many genes of interest. Expression level for most of the genes can be measured directly and be modified by involvement of different regulatory mechanisms, in particular, by the presence of regulatory proteins or their binding sites in regulatory DNA. Investigation into the gene expression pattern for individual genes, either tissue- or other condition-specific, is not a problem any more.

However, in nature, not all condition- or TF-regulated genes are primary target genes for that condition or TF, respectively. The transcriptional response usually happens in several successive steps. At first, direct activation of a small number of

primary target genes occurs within about 30 minutes and contributes to the primary response. The protein products of these genes then activate other genes to produce a delayed, or secondary, response, and so on. Thus, even a simple regulatory trigger can cause a very complex change in the pattern of gene expression [12].

Fig. 2.7 Detection of hormone primary target genes using translational inhibitors. Gene 1 and Gene 2 are the hormone-regulated genes (require presence of an activated hormone receptor for transcription process to start), however Gene 1 is a direct target of the hormone, while Gene 2 is not. a) In the absence of hormone ablation treatment, Gene 1 and Gene 2 are not expressed; b) When the hormone is added, its nuclear receptor is activated thus inducing expression of Gene 1. Product of Gene 1 is protein B, which, in its turn, induces expression of Gene 2. As a result, products of the expression of both genes is detected (mRNA levels can be measured by real-time PCR or Northern blotting [74]); c) When hormone ablation treatment is accompanied with translational inhibition, protein B is not synthesized and does not induce expression of Gene 2. Therefore, mRNA B is produced and detected, while mRNA C molecules are absent. The conclusion is as follows: Gene 1 is a primary target gene because it is expressed in the presence of the translational inhibitor (for example, cycloheximide), and Gene 2 is a secondary response gene.

A common method for detection of primary target genes is using translational inhibitors, e.g. cycloheximide, which stalls the ribosomes on the transcript by blocking the elongation process [75]. The idea of translation inhibition is to stop synthesis of the primary-response proteins which may be involved into regulation of expression of other genes and thus induce a secondary response. However, if a gene is still expressed in the presence of the translational inhibitor, it is supposed to be a primary target. The scheme of primary target genes detection is illustrated in Fig. 2.7 by the example of hormonal regulation: in the presence of the translation inhibitor, the secondary response is blocked, and thus the primary response stage could be specifically distinguished.

2.3 Bioinformatics for Gene Expression Regulation Study

Transcription factors are the part of the system that controls the transfer of genetic information from DNA to RNA via the process of transcription. Specific binding of TFs to gene promoter areas is essential for initiation of transcription of most eukaryotic genes. TFs usually employ a variety of mechanisms to recognize their DNA target sites [76]. In the last few years, attempts have been made to describe these mechanisms by general sets of rules and associated models.

2.3.1 Prediction of transcription factor binding sites

Transcription factors usually have distinct preferences towards specific target sequences. Given a set of known binding sites with confirmed functionality, it may be possible to construct a model for describing the target sequence properties which could be used for *de novo* prediction of potential binding sites in genomic sequences.

The problem of *in silico* TFBS recognition is two-faced. At first, it is necessary to select an appropriate way to model binding preferences using experimental data. Second, it is necessary to develop methods to apply the models to find functional TFBSs in promoter sequences. Every year, dozens of new approaches for TFBS prediction are reported by researchers from all over the world, and some time later, other researchers perform comparative analyses to understand the state of the arts in the area. A comprehensive review of general strategies for TFBS recognition was recently conducted by Wasserman and Sandelin [13]. Additionally, an exhaustive comparison of different applications with the aim to estimate how accurately different

types of motifs can be recognized in sets of unaligned regulatory sequences was done by Tompa et al. [77]. However, for *de novo* prediction of TFBSs in genomic sequences (note that the problem is different from that of multiple sequence analysis), very few large-scale methods have been offered with corresponding TFBS databases.

Consensus sequence is a concept implemented in molecular biology in the recent decades, and it is very important for understanding the biological role of the DNA in the evolution [78]. For example, the consensus sequences are used for the identification of mRNA initiation and termination sites, analysis of the secondary structure of RNA, alignment of the multiple DNA sequences, and identification of the molecular patterns that occur outside of a preset frequency. It is the consensus sequence method what was the first proposed for TFBS recognition in 1991 [79].

The consensus sequence refers to a sequence of nucleotides that occur frequently but not universally in the set of examples. DNA consensus generation is accomplished by aligning the example sequences and assigning a consensus nucleotide letter for each position. The consensus nucleotide is the one which best represents the nucleotide composition in that position, i.e. the nucleotide with the highest frequency. It is the number of allowed mismatches what actually regulates the trade-off between sensitivity and specificity of the consensus sequence model: a consensus sequence that allows no mismatches will identify too few signals, while a consensus sequence that allows too many mismatches will identify redundant false positives. In addition, consensus sequences confer an information loss as some of the possible nucleotides from the original data may not be reflected in the resulting consensus. Consensus sequences are also affected by unusual example sequences.

Position frequency matrix (PFM) is also the nucleotide-based method, and, per se, is the extension of the consensus sequence method, where the set of aligned sequences is represented by a matrix of $5 \times N$ or $4 \times N$ size. Every row of the PFM corresponds to a single nucleotide position within the motif, and N is the length of the motif pattern being considered. A PFM element represents the relative frequency of a particular nucleotide in a given position, while the number of PFM columns (either 4 or 5) depends on whether a gap is allowed for the preceding sequence alignment.

In 1995, Quandt *et al.* [80] proposed a versatile model for PFM-based TFBS recognition. The approach is since referred to as the *position weight matrix* as it implies calculation of matrix weights that are proportional to the information content

of every single nucleotide position within the TFBS pattern. Every element of the calculated PFM should be multiplied by the weight value for its nucleotide position, and the resulting matrix is then called a Position Weight Matrix (PWM), or Position-Specific Scoring Matrix (PSSM). Multiplying each nucleotide frequency by the nucleotide position weight value emphasizes the fact that mismatches at less conserved positions are more easily tolerated than mismatches at highly conserved positions. The idea makes sense since it has been shown that for a large and representative collection of binding sites, the overall information content of the resulting PFM is proportional to the actual binding energy [23].

Although there are certain inherent problems in using PWMs [25], the weight matrices are catered for good approximations of underlying patterns and result in useful easily interpretable motif representations [81]. A number of experimentally defined TFBSs have been assembled into publicly available collections of PWMs; the most popular two are TRANSFAC [82] and JASPAR [83]. The widely known tools MatInspector [84] and P-MATCH [85] use PWMs for TFBS prediction. The PWMs are also often used as parts of multiple-feature TFBS prediction frameworks. Additionally, using TRANSFAC, a first attempt to model tissue-specific regulation has recently been described [86].

The profile of the motif of interest could additionally be represented in the form of a sequence logo [87], which shows the relative frequency or weight of each nucleotide at the particular position in a sequence. This method confers no nucleotide information loss and enables a fast visual assessment of the consensus patterns. The nucleotide position methods are outlined in Fig. 2.8.

Markov chain, named after its inventor Andrei Markov, is a discrete-time stochastic process, which could be used to represent DNA sequences using *Markov models* [88]. Though Markov models can be of relatively simple architectures, it is also possible to define complex schemes, where Markov transition probability would non-trivially depend on the position within the modeled sequence. Parameters of the Markov models are usually estimated by the Maximum Likelihood approach, where probability of an event is associated with the frequency of that event observed while training. The approach is in some way similar to the PFM-based modeling, but the single nucleotide frequencies as atomic entities of analysis are replaced by inter-nucleotide transition probabilities. An example of successful using of the Markov

modeling for detecting signals in genomic sequences was described by Rajapakse and Ho [89].

A. Aligned training sequences

```

-gccagcaggtctgtgtccctga--
-acagatctcattctgttcccg--
-atagggtctcttgagtagtcta--
-aaaggctcttttcagttctagg--
-ctgtggtcagccagttctcag--
-gccagcaggttgtgtccctga--
-gccagcaggtgtgtccctga--
-gccagcaggtatgtccctga--
-gccagcaggtgtgtcccgga--
-gccagcaggtgtgtccctaa--
.....

```

B. Position matrices

Position frequency matrix (PFM)

position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
A	11	14	1	12	3	14	31	11	36	23	20	16	8	0	0	6	0	1	13	16	15
C	4	5	13	3	6	3	8	31	9	9	10	26	2	0	2	8	62	8	7	4	9
G	7	3	21	19	32	7	4	2	6	12	11	7	1	65	0	1	1	2	3	11	9
T	20	20	8	19	13	30	11	11	4	11	14	6	54	0	63	50	2	54	22	12	9
IUPAC	N	t/a	g/c	n	G	T	A	C	A	n	n	c/a	T	G	T	T	G	T	t/a	n	n

Position weight matrix (PWM)

position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
A	0.03	0.05	0.00	0.02	0.01	0.05	0.11	0.04	0.18	0.02	0.01	0.04	0.07	0.00	0.00	0.04	0.00	0.01	0.04	0.03	0.00
C	0.01	0.02	0.06	0.01	0.03	0.01	0.03	0.12	0.04	0.01	0.01	0.06	0.02	0.00	0.03	0.06	0.86	0.07	0.02	0.01	0.00
G	0.02	0.01	0.10	0.04	0.14	0.03	0.01	0.01	0.03	0.01	0.01	0.02	0.01	1.00	0.00	0.01	0.01	0.02	0.01	0.02	0.00
T	0.05	0.08	0.04	0.04	0.06	0.11	0.04	0.04	0.02	0.01	0.01	0.01	0.48	0.00	0.87	0.36	0.03	0.50	0.07	0.02	0.00

C. Sequence logo

Fig. 2.8 The frequency-based TFBS modeling paradigm. For the set of aligned sequences (A), the frequency of each nucleotide is stored into the Position Frequency Matrix and Position Weight Matrix (B). The consensus sequence is encoded using the IUPAC nomenclature [90]. Sequence logo (C) shows the relative weight of each nucleotide at each position in the sequence (generated by the free R package SeqLogo for the set of HREs described in the next parts of the thesis).

Though the PWM method is currently the most popular for TFBS modeling, there exist some other methods for DNA motif recognition. *Hidden Markov Models* [91] are used when there is no one-to-one correspondence between the sequence nucleotides and the states of the Markov model. However, for hidden Markov models, it is very difficult to design an appropriate set of states for modeling short DNA sequences like the TFBSs.

Gibbs Sampling can be used for recognition of motifs within a given set of aligned or unaligned sequences [92]. The method of Gibbs sampling exploits the

Monte Carlo procedure cycled in a recursion (the idea comes from the Bayesian inference about Maximum A Posteriori probability [93]). Biological background for the necessity of motif detection in multiple sequences comes from the assumption that co-regulated genes may have common TFBSs in their promoter regions. The Gibbs sampling method has recently become especially useful, since the microarray methods now provide a bioinformatician with enough data about co-expressed genes and genes regulated exclusively [94].

Involvement of the “beyond TFBS” characteristics of the promoter regions may provide access to better TFBS prediction results. In the works by Murakami et al. [95] and Pudimat et al. [96], several potential factors which define the context-based functionality of regulatory motifs are considered, namely, the tendency of TFBSs to cluster densely in small confined areas (for formation of transcription initiation complexes, that is how it works *in vivo*), the presence of CpG islands, the relative position within the promoter region, the structural properties and conformational parameters of the DNA (such as helical twist, helical slide, minor groove width), physical parameters (such as free energy, melting temperature), and even the corresponding gene expression profile. The DNA deformation energy has also been proposed as an indirect recognition mechanism for protein-DNA interaction and can be involved as an additional selection factor for predicting functional regulatory regions [97]. Though such advanced approaches are still based on quite a number of assumptions, they allow bringing out the future work directions for both *in silico* models for gene expression regulation, and *in vitro* experiments for gaining the reliable background information.

Since the first consensus sequence approach to TFBS recognition was reported in 1991, the *in silico* methods have evolved to complex algorithms which include phylogenetic footprints [98], gene expression data analysis [99], Bayesian trees [100] and graphs [101], and have already incorporated the biochemical properties of DNA and proteins [102]. Despite all above, recent reviews conclude that though the problem of binding site detection is mostly solved in terms of sensitivity [15] (that is, if the motif exists it will probably be found), current computational approaches usually fail to demonstrate appropriate specificity and still cannot be comparable to experimental methods.

2.3.2 Computational models for steroid hormone response elements

To date, no methods or tools for *in silico* analysis of progesterone-induced genomic and/or cellular effects have been reported. However, considering a wider problem, a few attempts to model dimeric binding sites as a special case of DNA motifs have already been performed.

In particular, the idea of investigation into hormonal regulatory activity using methods of computational biology is now being developed by the research groups in the Genome Institute and Institute for Infocomm Research in Singapore for the steroid hormone estrogen. One of the first works on studying the specific HRE-like patterns was reported by Bajic *et al.* [24]: the Dragon ERE Finder tool was developed for recognition of estrogen response elements (EREs), and further, an enhanced multi-platform system for ERE recognition has been developed on basis of its algorithm [103]. The proposed ERE recognition model included PFMs for every half-site and a matrix of transition probabilities between them. Lacking exhaustive training data, the Dragon ERE Finder though returned surprisingly high values for sensitivity and random expectation of ERE prediction. For example, with the sensitivity value of 83%, the reported random expectation was as high as 1 prediction per 13300 nt. Additionally, another tool by the same research group was developed for the sophisticated discovery of the estrogen-responsive genes [104].

In several papers, statistic models for recognition of response elements for the entire superfamily of nuclear hormone receptors (NHRs), which are known to bind their target DNAs as dimers, are proposed. One of the pioneering tools for NHRs' binding sites recognition is NUBIScan which uses position frequency matrices specifically adapted for the problem of recognition of dimeric motifs [105]. In a more recent work, Sandelin and Wasserman [106] also proposed a model for recognition of. The publicly available tool NHRScan exploits a hidden Markov model where the hidden states are the different dimeric structures of the partially repeated response elements, namely, the direct, palindromic, or inverted palindromic repeats considered as opposed to the neutral DNA. Each hidden state emits nucleotide bases guided by its own probability distribution. The model allows varying the HRE spacer length, and includes simultaneous screening of the complementary DNA strand. The main limitation of the work that the authors intended to resolve was that the training set consisted of only 107 functional nuclear receptor response elements, and it is indeed a

challenging limitation given the diversity of nuclear receptors. As a result, the prediction sensitivity plateau was found to be at the level of nearly 85% correctly labeled nucleotides for the direct and palindromic HREs, and of nearly 75% for the HREs with inverted dimeric structure.

In the work by Favorov *et al.* [107], a model for recognition of symmetrically structured DNA motifs, namely, the inverted or direct repeats or spaced motif pairs, in a set of unaligned DNA sequences was proposed. The authors created a specialized tool for finding weak motifs with spacers of unknown length, designed a probabilistic model, and described an optimization procedure. The model exploited the Gibbs sampler method. For testing purposes, the authors considered two transcription factors whose binding sites are notoriously divergent and difficult for computational modeling, namely, ArcA and NarP (both involved in the regulation of respiration). As a result, a number of genes possibly regulated by ArcA-P in *E.coli* and three other gamma-proteobacteria were predicted.

In the work by Bi *et al.* [108], a model for prediction of dimeric (so called “two-block”) TFBSs in DNA is proposed. The authors exploit Gibbs sampling approach and test the developed application using HNF4alpha, CAR/RXR and RXR/RXR nuclear receptors.

In conclusion to the section, steroid hormones, hormone receptors, and HRE-like DNA patterns are of intense interest for at least several bioinformatics research groups to date. For recognition of HRE-like motifs, different computational models were used and some interesting preliminary results have already been obtained. However, the field of computational regulatory genomics, including that associated with modeling of hormonal regulation, is being developed sweepingly. Accumulation of biological information opens more directions for further research.

2.4 High-Performance Computing Technologies for Machine Learning

As a broad subfield of artificial intelligence, the machine learning is concerned with the design and development of algorithms and techniques that allow computers to “learn”. The major focus of machine learning research is to extract information from data automatically by computational and statistical methods. Hence, machine learning is closely related to data mining, statistics, and theoretical computer science.

Nowadays, very large data sets are already common in many applications, such as Web mining, bioinformatics, speech recognition, and image processing. With advances in computer technology, we currently have the ability to store and process large amounts of data, as well as to access it from physically distant locations over computer networks. As the amount of available data increases, many machine learning algorithms become unable to process it using a single general purpose processor, and that is why high-performance technologies are becoming increasingly popular. In particular, multiprocessor and multicore architectures have already become widespread. Thus, it is important and necessary to adapt existing machine learning techniques for execution in parallel using distributed architectures in order to utilize the available computational resources and reduce processing time. Using application-specific hardware acceleration platforms to speed up certain stages of computations is an alternative approach for getting faster solutions than the conventional general purpose architectures may provide.

2.4.1 Machine learning

The goal of machine learning is to program computers to use example data or past experience to solve a given problem. This definition covers a broad range of learning tasks, such as how to design autonomous mobile robots that learn to navigate from their own experience, how to data mine historical medical records to predict patients' response to different treatments, and how to build search engines that automatically customize to their users' interests. To be more precise, we say that a machine learns with respect to a particular task T , performance metric P , and type of experience E , if the system reliably improves its performance P at task T following experience E . Depending on how we specify T , P , and E , the learning task might also be called by names such as data mining, autonomous discovery, database updating, programming by example, etc. [109].

Machine learning can be considered as an intersection of computer science and statistics, but in general it is none of them. While computer science is focused primarily on how to manually program computers, machine learning investigates into the question of how to make the computers to program themselves. While statistics looks for conclusions that could be inferred from data, machine learning incorporates additional questions about what computational architectures and algorithms can be used to effectively manage these data. Additionally, the third field whose primary

question is closely related to machine learning is the study of human and animal learning in psychology and neuroscience. Other fields, ranging from biology to economics and control theory, also have a fundamental interest in the problems of how systems can automatically adapt or optimize themselves in a given environment, and machine learning is expected to have an increasing exchange of ideas with these fields over the coming years.

There exist three major learning paradigms for machine learning, each corresponding to a particular abstract learning task [110]:

- **Supervised learning.** In supervised learning, we are given a set of example pairs $(x, y), x \in X, y \in Y$, and the aim is to find a function f in the allowed class of functions that correctly matches the examples. The cost function is usually associated with the mismatch between the resulting mapping and the actual data. The approach is usually used for the purpose of prediction and classification.
- **Unsupervised learning.** In unsupervised learning, we are given data x , and a cost function to be minimized. The cost function can be any pre-defined function of x and the network output. Most applications of the unsupervised learning strategy come from data mining, data processing, and data filtering.
- **Reinforcement learning.** In reinforcement learning, the data to process iteratively arises as a result of model's interaction with the environment. At each moment t , the model performs an action y_t while the environment generates an observation x_t and an instantaneous cost c_t , according to some (usually unknown) dynamics. The aim of reinforcement learning is to establish a strategy for selecting model's actions that minimize the long-term cost. Tasks that apply the paradigm of reinforcement learning are the control problems, games, and other sequential decision making tasks.

The measure of actual progress in machine learning is the applicability of machine learning approaches to important real-world problems. Indeed, many successful applications of machine learning methods already exist, including systems that analyze past sales data to predict customer behavior, recognize faces or spoken speech, optimize the robot behavior so that its mission can be completed using minimum resources, and extract knowledge from bioinformatics data. However, even

more problems still expect the machine learning solutions to be developed for them.

2.4.2 Reconfigurable computing

Over the past decades, we have seen an explosion of life science research and an exponential increase of gained biological data. At earlier stages, all the experimental data could be easily interpreted on-site, but since the high-throughput experimental methods have become available, the amount of information fell beyond the capabilities of manual examination, thus leading to establishing a new field of bioinformatics. Nowadays, computational capacities of modern computers have also become insufficient for many bioinformatics problems which involve genome-wide sequence analysis, long sequence alignment, multi-factor models, etc. Thus, modern bioinformatics have recently started to evolve towards the fields of high-performance computing technologies.

High-Performance Computing (HPC) is an integrated computing environment for solving large-scale computationally demanding problems in science, engineering, and business. Newly emerging areas of HPC applications include medical sciences, transportation, financial operations and advanced human-computer interface such as virtual reality. High performance computing includes computer hardware, software, algorithms, programming tools and environments, plus visualization.

There are two general ways to build high performance systems. The first is to exploit several computational processors, thus forming a computer cluster, and gain the computational speed due to resource distribution and parallelization. The second way is to use hardware-acceleration technologies to benefit from application-specific design which usually allows eliminating a bulk of redundant operations. The former is also referred to as grid computing and is already a common practice for time- and memory-consuming operations like multiple sequence alignment by BLAST [111], while the latter implies the usage of application-specific integrated circuits (ASICs) which may perform a pre-defined set of calculations faster than general purpose processors.

An application-specific integrated circuit is a circuit customized for a particular use rather than intended for a general-purpose execution. For example, a chip designed solely to run a cell phone is an ASIC. Application-specific circuit is the solution for high-performance or low-power system since it implies efficient, though less-flexible (or non-flexible at all), implementation of specific applications. Intermediate between

ASICs and general purpose processors are the application specific instruction set processors (ASIPs).

The ASICs can be fabricated by several technologies, including full-customization by specifying the layout of each individual transistor and all interconnections between them [112]. Full customization usually guarantees the best possible performance, but its practical applicability is limited due to high development cost in comparison with other hardware design approaches. In the standard design cell technology, a cell is a group of transistors and interconnection structures, and it is the locations and interconnections between these groups what is customized. Besides, the field-programmable gate array (FPGA) is the technology in which all the transistors are physically grouped into logic blocks during the manufacturing process and then no mask layers are customized, but it is the interconnections between the logic blocks (or gates) that is managed [113]. The main advantage of the FPGA in comparison with the other technologies is that due to no customization of the mask layers, the device can be re-programmed by the customer right in the field and as many times as required. Currently, the FPGA technology of hardware acceleration is becoming increasingly popular for many research areas due to its excellent combination of usability in the field and capability to outperform the computational speed of general purpose processors.

An FPGA chip is a semiconductor device that contains programmable logic components (called "logic blocks") and programmable interconnects. A hierarchy of programmable interconnects allows logic blocks to be connected as requested by the system designer.

The historical roots of FPGAs are in complex programmable logic devices (CPLDs) of the early to mid 1980s. Ross Freeman, Xilinx co-founder, invented the field programmable gate array in 1984 [114]. First CPLDs had the structural densities ranged from the equivalent of several to thousands of logic gates, while modern FPGAs typically range from tens of thousands to millions. Another important achievement of the application-specific chip evolution is the presence of the higher-level embedded functions (such as adders and multipliers) and the embedded blocks of memory in most of the modern FPGAs.

Nowadays, however, a major trend is to take the coarse-grained architectural approach as a step further by combining the logic blocks and interconnects of traditional FPGAs with embedded microprocessors and related peripherals to form a

complete "system on a programmable chip". Examples of such hybrid technologies can be found in the Xilinx Virtex-II PRO and Virtex-4 devices, which include one or more PowerPC processors embedded within the FPGA's logic fabric. An alternative approach to using hard macro-processors is to make use of "soft" processor cores that are implemented within the FPGA logic. Additionally, some modern FPGAs already have the ability to be reprogrammed while running, and this is already closely related to the idea of actual machine learning considered in the form of reconfigurable computing device that reconfigures itself to suit the task at hand.

To define the behavior of the FPGA, the user develops a programming code using a hardware description language (HDL) or a schematic design [115]. Common HDLs are VHDL and Verilog. Then, using an electronic design automation tool, a technologically mapped netlist is generated. The netlist can then be fitted to the actual FPGA architecture using a process called place-and-route that is usually performed by the FPGA manufacturer's proprietary software. The user validates the map and place-and-route results via timing analysis, simulation, and other verification methodologies. Once the validation process is complete, the binary file (also generated using the manufacturer's proprietary software) is used to (re)configure the FPGA chip.

Attempting to reduce the complexity of FPGA development using HDLs, which in fact are similar to the low-level assembly languages, certain moves have been performed with the aim to raise the abstraction level of the design. Software companies such as Celoxica are promoting special languages SystemC and Handel-C as a way to combine high-level languages (usually C or C++) with concurrency models essential for the hardware operation in order to allow faster design cycles for FPGAs than those achieved using traditional HDLs. To simplify the design of complex systems in FPGAs, there exist libraries of predefined complex functions and circuits that have been tested and optimized to speed up the design process. These predefined circuits are commonly called IP cores, and are available from FPGA vendors and third-party IP suppliers (rarely free, typically released under proprietary licenses).

Though the area of exploiting the FPGA technology for the problems of bioinformatics is very novel, it has already attracted certain attention of both bioinformaticians and hardware engineers. Since mid-2000's, conferences and symposia with presentations and discussions of hardware-accelerated solutions in bioinformatics are held over the world, and although there is still no specialized journal for the area, both bioinformatics and engineering journals already consider

special issues focused on the combined effort field.

Several useful applications of the FPGA technology to the area of bioinformatics have already been reported with the bioinformatics algorithms being implemented as FPGA co-processors for the flow of data analysis. The idea of using high-performance application-specific co-processors has already been exploited in other applications; perhaps the most commonly known one is the real-time graphic data processing, where the so called "graphics cards" carry processors and memory dedicated to the computationally intensive image rendering. During the processor-to-co-processor cooperation, general-purpose CPUs transfer massive processing tasks to the specialized co-processor, and process the results collected from that co-processor.

Since many bioinformatics algorithms require repetition of a huge number of the similar processing steps on different sets of data, the massively parallel and adjustable-to-application computational capacities of FPGAs seem to be offering a viable, readily available solution when added to the host computer's general purpose processor as an acceleration co-processor. The pioneering examples of the "bioinformatics on-chip" are the hardware-accelerated solutions for the famous database search tool BLAST [28, 116], distributed hardware acceleration of the popular hidden Markov model-based search tool HMMer [27], and the program for the phylogenetic tree inference RAxML [117]. For the problem of molecular dynamics modeling, a 57-fold superiority of the FPGA implementation over a software application run on a conventional PC has recently been reported [118].

Although FPGAs have been around for nearly two decades, their application in bioinformatics is only becoming a common practice. However, several basic bioinformatics methods, namely, HMMer, Smith-Waterman, and ClustalW, are already available in hardware as commercial solutions, and this assures that the next-generation of on-chip biology research is even more exciting.

Part II

Additive Statistic Model of Hormone Response Element

Chapter 3

Position Frequency Model for Generation of the Consensus HRE Sequence

In this chapter, we present a position frequency-based model of the HRE consensus, and use this model to recognize HREs in DNA sequences. The “position frequency” definition for the model stands for the assumption that mono- or oligo-nucleotides are more likely important for intermolecular binding if they are found in most of known binding sites, while the mutations which are rarely (or never) seen in functional protein-DNA complexes are assumed fatal.

In Section 3.1, we present a preliminary study on experimental work aimed at elaboration of the HRE structure, mainly by mutational analysis. A general PWM model of the HRE consensus is introduced in Section 3.2. This model (and other models described in this thesis) is trained by a collection of experimentally validated HREs, which is also introduced in the section. Using this collection, a homogeneity test is performed for verification of the assumption that the receptors for the three steroid hormones, namely, androgen, glucocorticoid, and progesterone, indeed share the same response elements in DNA, as it has been reported earlier in several molecular biology papers [21, 119, 120]. The results of *de novo* HRE prediction by the PWM-based method using the first-order and higher-order PWM models are described in Section 3.3. In particular, the cases of di- and tri-nucleotide weight matrices are considered besides the conventional mono-nucleotide PWM.

3.1 Understanding HRE Structure through Mutation Analysis

In natural promoters, steroid HREs display varying diversity in their DNA sequences. Some nucleotide positions may contribute to the degree of receptor specificity or binding affinity, whereas others may be incidental [58]. The HRE consensus sequence is believed to have been derived from an ‘ancestor’ sequence from which all mutations arose [121] but due to the loss of information when the consensus sequence is reconstructed, the evolutionary relationship between the sequences becomes obscured. The general idea of HRE acting as a DNA dimer has been known

for years, but only thorough mutation analysis might help to explore the tiny composition of a functional response element. In this section, we investigate into the research works concerning preferences of hormone receptors towards their target DNA sequences.

In the work by Dahlman-Wright *et al.* [64], a general form of the HRE structure was studied. Using glucocorticoid receptor EMSA, it was found that replacement of the right half-site of the HRE reduced its binding affinity significantly. The length of the inner spacer between the half-sites was also found to be an important factor. When the spacer was 3bp long, the portion of DNA molecules formed dimeric protein-DNA complexes was as high as 40%, in contrast to that of other length values: the 2bp spacer resulted in 5% of DNA bound to the receptor proteins, and the spacer length of 4bp gave less than 4% of bound DNA molecules. Even inversion of the half-site's orientation reduced binding properties of the response elements from 40% for the consensus to 10% for the reverse orientation of the half-sites.

Barbulescu *et al.* [122] performed mutation analysis of two functional AREs from mouse *Pem* gene (reproductive homeobox 5) promoter. Mutations of the right half-site and changes of the spacer length decreased binding properties of these AREs significantly (according to the EMSA results), while several point mutations in the left half-site were tolerated with high level of admission.

Single nucleotide mutation analysis was performed by Truss *et al.* [123]. Influence of substitutions of thymines in the right half-site was measured by EMSA using the progesterone and glucocorticoid receptors. The observation was as follows: in the 5'-TGTTCT-3' consensus (to which 50% of DNA probes bound), the thymine in the third position was of greatest importance: when replaced it reduced the HRE binding affinity to 15%. Additionally, the first thymine was less important (30%), while the substitution of the last one was easily tolerated (45%). Similar results were reported by Thackray *et al.* [124] for progesterone using EMSA: the second guanine and the third thymine played a crucial role for protein-DNA binding, while the other nucleotides of the right half-site were less important, and nucleotides from the left half-site were even more flexible. These results are not surprising if we also consider the work by Luisi *et al.* [71], where crystallographic analysis of GR-GRE interaction is reported. In particular, it is confirmed that the guanine at the second position in the right half-site (together with the fifth cytosine in the same half-site, and guanine and

cytosine from 2nd and 5th positions in the left half-site respectively) are the major contact points for the bound receptor.

Mutation analysis of the ARE was also performed by Roche *et al.* [125], Haelens *et al.* [126], Zhou *et al.* [127]. These researchers performed selection assays to identify AREs among large sets of randomly generated oligonucleotides. The first work by Roche *et al.* presented a list of collected AREs left after four rounds of assay selection. As expected, the results were quite similar to those known previously, namely, the 100% conservation of GpT in the right half-site, high conservation of the rest of the right half-site, relatively high flexibility of the left half-site with exception of the contact points, and a uniform nucleotide distribution within the HRE spacer and flanking regions. However, Haelens *et al.*, in addition to similar observations concerning the ARE composition, have found that the flanking regions and the spacer also play active role in binding. In the work by Zhou *et al.*, after six rounds of selection it was found that the spacer and flanking regions were also preserved rather than consisted of uniformly distributed nucleotides.

Progesterone response element (PRE) was thoroughly analyzed by Lieberman *et al.* [60] using both EMSA and CAT transfection assay. Each single nucleotide was changed to all the possible variants, and the respective PRE binding affinity was measured in two experiments. Additionally, analysis of a single nucleotide insertion into the spacer was included into the list of investigated PRE mutations. The findings showed that, for the right half-site 5'-TGTTCT-3', the second G, the third T, and the fifth C were of greatest importance for PRE binding properties. Assuming the affinity level of the PRE consensus to be 100, replacement of the second G with A resulted in the affinity level of 6, with C in the affinity of 9, and with T in the affinity of 2. Replacing the third T with A resulted in the affinity of 51, with C in the affinity of 6, and with G in the affinity of 12. Similarly, replacing the fifth C with A, G, and T resulted in the relative binding affinity of the PRE of 7, 3, and 17, respectively. In addition, the presence of the 3bp-long spacer is also important: a single nucleotide insertion decreased the relative affinity to the level of 9. On the other hand, the first and the fourth T's demonstrated less conservation: when replaced with A, C, and G, they returned the relative binding affinities for the PRE of level of 59, 94, and 34 for the former, and of 48, 54, and 66 for the latter, respectively. The last thymine nucleotide seemed to be not important at all because, when changed to adenine, it resulted in the binding affinity of 30, to cytosine in the affinity of 118, and to guanine

in that of 72. Considering the left-half site, its cytosine in the 5th position was found to be essential for protein-DNA interaction (when replaced with A, it returned the zero binding affinity, with G – the affinity of 25, replacement with T was not examined), while other nucleotide positions within that half-site were nearly as flexible as the spacer. In particular, after single nucleotide substitutions in the left-half site, the PRE binding affinity decreased to the level of 80 on average, and the lowest value was 24 for guanine at the 2nd position in the consensus 5`-TGTACAnnnTGTTCT-3` replaced with thymine.

One of the most exhaustive analysis of ARE, PRE, and GRE was done by Nelson *et al.* [61]. This group performed selection assays for identification of response elements with both high binding affinity and high receptor specificity for the three steroid hormone receptors. The only important limitation of this work was that no flexibility of the HRE right half-site was allowed. Although this assumption has a basis as the right HRE half-site is indeed known to be well conserved, this half-site still admits some single nucleotide substitutions in non-contact points [125].

In short, the functional HRE sequences display both conservation and diversity, and the latter makes it a challenging task to establish a computational model for reliable HRE recognition.

3.2 A General Position Frequency Model

Based on the analysis on the previous computational methods, in this section, we present a position frequency-based statistic model for HRE recognition and an HRE dataset further used for training and testing purposes.

The goal of construction of a HRE sequence database is to allow effective training of the statistic models. For the case of HREs, it is of great importance that the size of the training set is large enough in order to capture the possible variations of the HRE sequence. The diversity of functional HREs is described in the previous section, so a small training set size would tend to generate redundant false negative outcomes [128].

For our collection, only the HRE sequences validated through experimental methods are used. The collected HREs should also be accompanied with the relevant information such as the method used for validation, the organisms from which sequences is obtained etc. The collection will be then used for training the PWM

model and other HRE recognition models described in this thesis.

3.2.1 Construction of the HRE training database

Accuracy of a statistic model largely depends on construction of the training dataset. One can easily achieve very high sensitivity and specificity with just a few training sequences, but this result would tend to be meaningless because the relative variance and the corresponding reliability will also be very high.

There exist a few public databases of TFBSs and *cis*-regulatory modules, among them are TRANSFAC [82], JASPAR [83], PReMod [129]. However, there is little coordination between these databases, resulting in duplicated data that needs to be sifted through in order to avoid errors. Additionally, none of these databases house exclusively HREs or other nuclear receptor response elements. By scanning through these public databases, we found nearly 50 HREs with reliable annotation. In particular, the publicly available version of TRANSFAC contains the GRE matrix calculated using 38 binding sites, and JASPAR has the ARE weight matrix with 24 sequences. Genomatix TFBS database, which is a part of the MatInspector TFBS prediction tool [84], can be used for predicting ARE, GRE and PRE sequences separately, but this database is constructed based on only one experimental result [61] with a strong limitation for HRE selection process. That is, a reliable and exhaustive (but non-redundant) dataset of HREs, which might be used for training and testing purposes, is a must for further development of any HRE recognition model; otherwise, any possible experimental results will be inconclusive.

For this purpose, a set of DNA sequences of experimentally verified hormone receptor binding sites was collected from literature sources. At first, we collected PRE only, but this hormone looked like to be the least investigated among all steroid hormones, so we considered a canonical assertion that glucocorticoid, androgen and progesterone receptors tend to share the same response elements on DNA. The collected ARE, PRE, and GRE sequences are housed in our database Tiger HRE DB [130].

Selection of HRE sequences

Almost 700 experimentally verified binding sites for androgen, glucocorticoid, and progesterone nuclear receptors were collected from more than 200 biomedical

literature sources. The latest version of the database contains 66 progesterone response elements (PREs), 377 glucocorticoid response elements (GREs), and 218 androgen response elements (AREs). For a response element to be accepted into the collection, a convincing experimental evidence was required:

- the HRE was validated for binding to its receptor *in vitro* by any experimental method (EMSA, protection assay, etc.), or
- the HRE was shown to mediate the regulatory response through transfection assays (usually with a CAT reporter plasmid).

In the latter case, identification of the interacting receptor protein and the experimental validation of the binding site position should have been presented. The HRE sequence was not included into the collection if the literature source contained ambiguous or insufficient information; in particular, if the experimental data confirmed only approximate location of a long protected region and then the possible positions of the receptor binding sites were predicted by sequence analysis. Overfitting of the model was avoided by limiting the collection to only one sequence for a particular HRE in the database, even if multiple experimentally verified observations existed for a primer taken from one source. The reported bound sequence was included into the database with five flanking nucleotides in both directions. Positions of the two half-sites of the response element were also recorded if that information was provided.

All retrieved binding sites are stored in our Tiger HRE database. In this database, each entry is characterized by:

- The response element nucleotide sequence (with the positions of the two half-sites to which the receptor binds as a dimer, if known),
- The steroid hormone for which the receptor binding was confirmed (if the same binding site is reported to bind to two or three steroid hormone receptors in the same source, it corresponds to separate database entries),
- The corresponding hormone-regulated gene (if applicable; note that a large part of the collection is taken from selection assays performed without reference to particular genes),
- The organism from which the genomic DNA used in the experiment was

retrieved,

- The relative position from the transcription start site (if this response element is from a promoter or an enhancer region or the first exon of any hormone-regulated gene),
- The experimental method of DNA-protein complex identification, and
- The reference to the literature source.

The database is implemented using the MySQL relational database system. An SQL query result of “SELECT hormone, seq, gene, snms, species, reference, method FROM literature” is shown in Fig. 3.1.

hormone	seq	gene	snms	species	reference	method
Progesterone	CTCAGAACAGGCTGTTTCAGGG	ABCG2	0.801	HS	[85]	EMSA
Progesterone	TGCTGTACAGGATGTTCTAGC	ABCG3	0.982	HS	[85]	EMSA
Progesterone	TGGCACACAAAATGTGCAGCC	CCND1	0.838	HS	[86]	EMSA
Progesterone	AGCAGCATATGAAGTCCAAGA	CCND2	0.785	HS	[86]	EMSA
Progesterone	CACACATGAGTGTGTCCTAGT	EHF	0.869	HS	[87]	EMSA
Progesterone	AGAACAGAGTGTTCT	EHF	0.986	HS	[87]	EMSA
Glucocorticoid	AGGCTGGAGAGACGTCCAGTT	FN5	0.705	MM	[88]	DNase
Glucocorticoid	AAGGAAAGGACCTGTTCAAGG	FN6	0.855	MM	[88]	DNase
Glucocorticoid	TGCCTTGTTTAATGTCCCCCA	FN7	0.801	MM	[88]	DNase
Glucocorticoid	TTAATTCCAAGAAGTCCACAT	FN8	0.772	MM	[88]	DNase
Glucocorticoid	CCAAGAAGGAAGTGTGTAGC	FN9	0.810	MM	[88]	DNase
Glucocorticoid	CTGCTCTCTGTGTGGCCAA	FN10	0.661	MM	[88]	DNase
Glucocorticoid	CATTCTCCCTTCTTCTCTCC	FN11	0.850	MM	[88]	DNase

Number of fields returned: 7

Fig. 3.1 A query from the Tiger HRE DB system.

However, due to the intrinsic limitations of the TFBS identification methods (refer to the Section 2.2), further processing of the collected HREs by the algorithm of sequence alignment is required.

Method for position-specific alignment

Although most of experimental methods allow identification of a DNA region which indeed contains an HRE with good accuracy, it is still difficult or impossible to

detect a response element with the single nucleotide precision unless sophisticated and expensive methods of structure analysis, like nuclear spectroscopy or crystallographic analysis, are involved for investigation of the protein-DNA complex. We propose a solution to resolve that limitation by accurate alignment of all collected HREs.

It is widely known that the canonical HRE consists of two half-sites separated by a spacer of several nucleotides. Single nucleotide mutations are usually tolerated in some positions within the half-sites, while other positions (which are the contact points involved into interaction with the DNA-binding domain of the receptor protein) have been found to be critical for binding affinity. It seems reasonable to use the information about this position-specific conservation for alignment of HREs. In particular, the mismatches in the conserved positions should be penalized higher than those in the non-preserved part of the HRE. Thus, we propose the idea of *position-specific alignment* for reconstruction of the HRE database. The invented algorithm for alignment is performed in three steps.

Step 1: Alignment of the preserved half-sites.

As the right half-sites of functional HREs are known to keep the highest conservation, they are used as “anchors” for determination of positions of the HRE half-sites within the experimentally confirmed HRE sequences. The consensus sequence of the right HRE half-site 5'-TGTTCT-3' is used for pairwise alignment with each experimentally validated HRE. Dealing with such short sequences, we are not forced to avoid greedy alignment algorithms, and the dynamic programming is used for semi-global alignment [131].

Even within the conserved half-site of the HRE, some nucleotides, such as guanine in the 2nd position of the half-site consensus, do not allow for any substitution, while mismatches in other positions are not so critical for the resulting binding affinity, as it has been shown for the 6th thymine. This criterion is used for calculating the substitution matrix and gap penalties for the alignment algorithm.

In the work by Lieberman et al. [60], each single nucleotide replacement in the HRE sequence was evaluated twice: in transfection assay experiment by measuring the corresponding CAT activity, and by quantification of bound complexes in a gel retardation experiment. In particular, the relative significance of each point mutation was characterized by the percentage metric, where the consensus was set to 100. The reported statistic data is similar for CAT and EMSA experiments in all nucleotide

positions within the HRE, so we conclude that this information reliably describes the significance of each mutation and can be used for reconstruction of required substitution matrices. Furthermore, we add the corresponding binding affinity values from these experiments, and use the average value $V(i,j)$ as a significance of the mutation from nucleotide base i to nucleotide base j . Thus, the substitution matrix $S = \{S(i,j)\}_{i,j=A,C,G,T}$ is defined as follows:

$$\begin{array}{ll} V(i,j) < 10: & S(i,j) = -3; \\ 10 < V(i,j) \leq 25: & S(i,j) = -2; \\ 25 < V(i,j) \leq 50: & S(i,j) = -1; \\ 50 < V(i,j) \leq 90: & S(i,j) = 0; \\ V(i,j) > 90: & S(i,j) = 1; \end{array}$$

where $S(i,j)$ is the corresponding element of desired substitution matrix, as in Table 3.1. The gap penalty is equal to the minimal mismatch penalty $g = -3$, in order to make it possible to replace the worst matching by a frame-shifting gap.

Table 3.1 Substitution matrix for position-specific alignment of HREs

	-7G	-6T	-5T	-4A	-3C	-2A	-1A	0A	+1C	+2T	+3G	+4T	+5T	+6C	+7T
A	1	1	0	1	-3	1	1	1	-1	-1	-3	-1	0	-3	-2
C	0	0	-3	-1	1	0	-1	1	1	0	-3	-2	0	1	1
G	1	1	1	-1	-2	1	1	-1	1	-2	1	-2	0	-2	-1
T	-1	1	1	-3	-3	-3	1	1	0	1	-2	1	1	-2	1

Then, while aligning a given sequence to the consensus sequence of the right HRE half-site, this substitution matrix (namely, its columns from +2 to +7) is used for evaluating their similarity. Therefore, our algorithm is advantageous over the conventional alignment algorithm since the mismatch penalty depends not only on the two compared nucleotides, but also on where they are located within the HRE sequence.

Step 2 Alignment of the non-preserved half-sites.

After all HRE sequences are aligned against right half-site, and hence anchored against each other, it is necessary to detect the locations of the other half-sites. The procedure of the second position-specific alignment is similar to the previous one, but

the significance of the spacer length between the two half-sites should also be taken into account.

Previous analysis of the HRE structure showed the optimal spacer length was 3 base pairs. When it was decreased or increased, the affinity of the binding site to its receptor protein DBD fell greatly. For example, Lieberman et al. [60], reported that in the case of a 4bp spacer, the relative CAT activity was as low as only 9% of a wild type, while the number of bound receptors was hundred times lower in comparison with its counterparty of the 3bp spacer. Similar result was reported by Dahlman-Wright et al. [64] for the cases of 2bp and 4bp spacers, where the proportion of bound DNA was 10 and 8 times lower respectively. Therefore, the procedure of position-specific HRE alignment is modified as follows: the score of each alignment is multiplied by GMScore (gap multiplying score) which is:

$$\text{GMScore} = \begin{cases} 1, & \text{if } \text{gap} = 3 \\ 0.1, & \text{otherwise} \end{cases} \quad (3.1)$$

This guarantees that the gap length is set to the value of 3 unless there are nearly exact consensus HRE half-sites with an additional single nucleotide insertion between them. The rest of the procedure of the half-site alignment is performed in the same way as described for the Step 1.

Step 3. Collecting the set of sequences

The sequences aligned against the consensus HRE are stored into the database of experimentally validated HREs. The pattern for the HRE is set to be 21bp-long, including the 15bp-long HRE consensus and 3bp-long flanking sequences on both sides. All nucleotides which exceed these bounds are cut off. A snapshot of the aligned GRE sequences is shown in Fig. 3.2. For use in the downstream computations, we also converted the HRE alignments into plain text format.

Using the set of aligned HRE sequences, we can calculate PFMs for the ARE, GRE, and PRE data sets separately. Additionally, PFMs can be calculated not only for the single nucleotide frequencies, but also for the oligo-nucleotide units of the sequences. For the case of multiple nucleotides, their positions are defined at their first bases.


```

glucocorticoid_2 - Notepad
File Edit Format View Help
0      ----gttaca---tggttct----
1      -aacgacacggagtcctttt--
2      -----tgtacagttct-----
3      -tttggtatcaaagtgttctgat--
4      -ctgagctcttagtggttctatt--
5      -tctattttcctatgtcctttt--
6      -atggttacaaactgttcttaa--
7      -ctctgttcactctgttctatg--
8      -ttggttacaaactgttct----
9      -atctgtacaggatgttctaga--
10     ----tgtacaggatgttct-----
11     -tttggtatcaaagtgttctgat--
12     -ctgagctcttagtggttctatt--
13     -tctattttcctatgttctttt--
14     -ttggttacaaactgttcttaa--
15     -atggttacaaactgttcttaa--
16     -tttggtatcaaagtgttctgat--
17     -ctgagctcttagtggttctatt--
18     -tctattttcctatgttctttt--
19     -atggttacaaactgttcttaa--
20     -tttggtatcaaagtgttctgat--

```

Fig. 3.2 Aligned GREs sequences.

Fig. 3.3 Relative frequencies of the nucleotide T along the HRE sequence for each of the three response elements and for the joint dataset of HREs.

Thus, with the set of the aligned HRE sequences, we are capable of making some initial observations about the HRE composition and comparing it with the existing

experimental findings (reviewed in Section 3.1). In particular, the relative conservation of certain nucleotide positions within HREs matches that found by molecular biology experiments. For illustration, the frequency distributions of thymines in the response element sequences are given in Fig. 3.3. The nucleotide T is used for illustration as the consensus HRE is known to be T-rich, hence the statistics is more obvious.

3.2.2 HRE homogeneity test with androgen, glucocorticoid, and progesterone

It is widely believed that AR, PR and GR tend to share the same binding sites on DNA, as it has been shown in many experiments. As this assumption will be the foundation of modeling the HRE consensus, we should confirm it by conducting the homogeneity test against the collected sequences. The basic idea is that AR, PR, and GR share the same binding sites if the frequency matrices computed for their response element sequences are statistically similar.

First, we compute the PFMs for each of ARE, PRE, and GRE sequence collections using the position-specific alignment procedure described in Section 3.2.1, and denote these matrices as S_a , S_p , and S_g , respectively. Each matrix is then considered as a collection of m rows, where m is the length of the HRE pattern. Each row corresponds to a single nucleotide position with the pattern, and is represented in the form of a 5-tuple (four components for nucleotide bases and one for the alignment gap).

Chi-square test [132] is a statistical criterion for checking the hypothesis that two or more independent series of observations come from the same probability distribution. We use this test for validating homogeneity of our HRE dataset in order to accept or reject the assertion that the three hormone receptors of interest share the same binding site on DNA. If so, the three binding sites are expected not to display statistically different nucleotide frequency compositions. The HRE homogeneity method operates as follows.

Let the independent series be the rows of HRE matrices S_a , S_p , and S_g , and each of these series consists of n_1 , n_2 , n_3 observations (HRE sequences), respectively. If these series are considered as experiments with a random variable, this variable can take one of four or five different values. The possible outcomes are the four letters of the DNA alphabet and the alignment gap, if applicable. The alignment procedure described in Section 3.2.1 implies the presence of a gap, so we consider the five-outcome experiment.

Let v_{ij} be the number of occurrences of outcome i in series number j such that $\sum_{i=1}^5 v_{ij} = n_j, j = 1, 2, 3, n_1 + n_2 + n_3 = n$. To check the hypothesis H_0 that all three HRE nucleotide patterns belong to the same frequency distribution, a statistical function belonging to $\chi^2((5-1)(3-1))$ distribution is computed as follows:

$$X_n^2 = n \left(\sum_{i=1}^5 \sum_{j=1}^3 \frac{v_{ij}^2}{n_j v_{ij}} - 1 \right) \quad (3.2)$$

The values v_{ij} and n_j should be large enough, so that the approximations done for calculation of this function are not violated [132].

After calculating the value of the statistical function, we check if this value belongs to the distribution $\chi^2(8)$ with the significance level α . If this is the case, we conclude that the HRE data do not contradict H_0 ; otherwise, the hypothesis of homogeneity should be rejected. After performing the homogeneity test for each of the m rows of matrices S_a , S_g , and S_p , we get a column of m values of the statistical function X_n^2 .

The results of the test with significance level $\alpha = 10^{-2}$ are shown in Table 3.2. The first column of the table is the nucleotide position in the HRE sequence. The second column is the value of the statistical function X_n^2 computed for this position using the rows of the three position frequency matrices as described above. The third column is the probability for a random variable with $\chi^2(8)$ distribution to be higher than the value of X_n^2 (we use one-sided test). If the X_n^2 value is high enough for the given significance level, we can conclude that the three considered random variables are not from the same underlying distribution; otherwise, we accept the H_0 hypothesis and claim that for the androgen, glucocorticoid, and progesterone response elements, their nucleotide frequency distributions are the same with respect to the collected data.

As expected, the most similar regions coincide with the highly conserved right HRE half-site (positions 13-18 in the table). Homogeneity p-values for this region are much higher than those for the rest of the HRE sequence, corresponding to less probable heterogeneity. The region of high conservation is highlighted in the resulting table below.

Table 3.2 Results of the homogeneity test for ARE, PRE, and GRE.

	Value of the statistical function, X_n^2	Probability of X_n^2
1	21.63	0.059
2	41.95	0.00023
3	25.88	0.039
4	36.27	0.0016
5	13.25	0.648
6	17.78	0.274
7	22.71	0.090
8	16.15	0.348
9	28.64	0.018
10	50.94	8.4E-06
11	25.68	0.041
12	26.13	0.037
13	23.63	0.071
14	1.19	0.999
15	6.01	0.979
16	15.04	0.448
17	6.93	0.959
18	9.77	0.833
19	35.62	0.002
20	3.06	0.999
21	38.54	0.00075

It is even more important that in the less conserved left HRE half-site the contact points 5 and 8 are also highly homogenous. The rest of the HRE pattern is slightly more heterogeneous among the three steroid hormones of interest, but after performing Bonferroni [133] and then less stringent Benjamini-Hochberg [134] multiple test corrections, none of the resulting p-values were found to be statistically significant. Therefore, it is legitimate to join the three position frequency matrices into one, and consider functional GREs and AREs as response elements for progesterone receptor. For further experiments in this chapter and below, the joint HRE dataset is used unless stated otherwise.

3.2.3 Position weight matrix model

A position weight matrix is a commonly used way to represent motifs in biological sequences. The PWM is a matrix of score values used to compute a

weighted similarity to the underlying motif pattern for a given sequence of a fixed length. The first-order weight matrix has one column for each symbol of the alphabet, and one row for each nucleotide (nt) or amino acid (aa) position in the pattern. Thus, the matrix element $Score(i,b)$ is the weight of b (nt or aa) at the position i of the motif sequence. For the input sequence, the weights of its nt or aa components are obtained from the matrix and then summed up, resulting in the weight of the entire sequence. The elements of the HRE PWM are computed as follows.

First, a joint HRE PFM is calculated using the aligned training sequences as described in Section 3.2.1. Each element of the PFM, $v_{i,b}$, is a relative frequency of the nucleotide b found in the position i of the HRE sequences from the training dataset.

Secondly, for weight calculations, the elements of information content vector $\mathbf{C}$, which is sometimes referred to as conservation score, are calculated by

$$C_i = \frac{1}{\ln(5)} \left(\sum_{b \in \{A,C,G,T,gap\}} v_{i,b} \times \ln v_{i,b} + \ln(5) \right), \quad (3.3)$$

where $i = 1, \dots, m$, and m is the length of the HRE pattern. Normalization coefficient is chosen to be equal to 5 in order to meet $C_i \in [0; 1]$. If the model of position-specific alignment without gaps is used, then the number of columns in the HRE PFM/PWM and the corresponding normalization coefficient should be set to 4.

Each element of the conservation vector C_i for HRE nucleotides is proportional to information content in that position, which, in turn, is indirectly concerned with nucleotide to amino acid binding energy [23]. Additionally, C_i takes value of 0 when nucleotide distribution in the position i of the HRE pattern is uniform and demonstrates no specific preservation. On the other hand, C_i becomes 1 only in the case of strong conservation on a particular nucleotide.

Thus, another illustration of relevance of the HRE collection is the distribution of conservation coefficients C_i . In particular, the information about relative positional conservation highly correlates with the existing data about the significance of point mutations. Even for the left half-site, which is much less conserved than the right one, the contact points on the 2nd (number 5 on the X axis) and 5th (number 8 on the X axis) positions are characterized with higher values of information content than others. Fig. 3.4 shows the distributions of conservation vectors $\mathbf{C}$ for all the three PFMs and the

summary matrix which is the matrix calculated from the joint dataset of HREs.

Fig. 3.4 Conservation coefficients for ARE, PRE, GRE PFMs, and for the joint PFM

Finally, the elements of the HRE PWM are defined by

$$Score(i,b) = C_i \times v_{i,b}, i = 1, \dots, m, b \in \{A, C, G, T, gap\} \quad (3.4)$$

Using this notation, for a given sequence of length m , its similarity to the HRE motif represented by the matrix is calculated as follows:

$$Similarity = \frac{\sum_{i=1}^m Score(i,b) - \sum_{i=1}^m \min Score_i}{\sum_{i=1}^m \max Score_i - \sum_{i=1}^m \min Score_i}, \quad (3.5)$$

where $\max Score_i = \max_{b \in \{A, C, G, T, gap\}} Score(i,b)$, $\min Score_i = \min_{b \in \{A, C, G, T, gap\}} Score(i,b)$, b is the i^{th} nucleotide of the sequence under validation, and $Score(i,b)$ is the element of the HRE PWM located in row i and corresponding to nucleotide b . The higher is the similarity coefficient, the larger are the corresponding $Score(i,b)$ values, and vice versa. That is, high similarity values correspond to the sequences composed of nucleotides frequently found in the training set of experimentally validated HREs.

As we have already proved that the PFM matrices for the three types of HREs are statistically similar (Section 3.2.2), we further evaluate the performance of the matrix-based HRE recognition methods using the joint position frequency matrix. This matrix is computed using the merged set of steroid HREs.

For implementation of the procedure of PWM-based HRE prediction, we at first align the DNA sequences of HREs from the experimentally validated collection using the position-specific alignment procedure described in the Section 3.2.1, and then calculate the resulting PFM. Second, we compute the HRE PWM using a part of the collection as a training set. Finally, we use the rest of the HRE collection and the set of neutral DNA as a testing set for *de novo* HRE prediction.

The algorithm for PWM-based HRE prediction works as follows. For a sequence of interest, a window of length 21 (equal to the height of the position weight matrix) shifts along, and for each of the subsequence its matrix similarity is computed using equation (3.5). Note that some of existing weight matrix methods may also involve the so called “core similarity” value, or similarity of the most conserved subpart of the binding site (which is usually the main contact region of length 4 [80]), but for the case of the dimeric binding sites this idea was found not to work well. Finally, if the matrix similarity value is larger than a pre-defined prediction threshold value β , then the sequence is marked as a putative HRE and stored into a database together with its annotation.

3.3 Extended Position Frequency Model for HRE Recognition

In most applications of DNA motif recognition, PWMs involve mono-nucleotide frequencies only for their basic calculations. However, some oligo-nucleotide patterns can also be over- or under-represented in comparison with corresponding products of independent frequencies of single nucleotides. The putative discrepancy implies that adjacent base pairs within genomic DNA are not independent from each other, and some nucleotides tend to co-exist, or, on the contrary, cannot be located next to each other. This trend can be found even in a whole-genome scale, where CpG dinucleotides are eliminated by the process of DNA methylation [135] and thus are much less frequent in real genomic sequences than it might be expected based solely on their GC content [136]. For the TFBS patterns in their general form, the presence of interdependent nucleotide effects on binding sites’ affinities was also confirmed

[26]. Therefore, it is reasonable to expect some oligo-nucleotide preservation within short sequences of HREs as well, so as to further justify the underlying position frequency model.

Let the order of a PWM model be the number of consecutive nucleotides considered as an atomic entity for PWM calculations. Conventional mono-nucleotide PWMs are thus used in the first-order models. In a higher-order PWM model, not the mono-nucleotide frequencies but the position frequencies of all possible oligo-nucleotide entities, namely di- or tri-nucleotides, are calculated.

In this section, we investigate into the order of the position matrix model, which is necessary and sufficient for HRE recognition, using the homogeneity test. We also test the prediction accuracy of the first- and the second-order PWM models.

3.3.1 Higher-order PWMs: correlation and justification

To justify the necessity of using di-nucleotide patterns for HRE recognition, we design and carry out the following experiments:

- First, we generate the so called *expected* di-nucleotide frequency matrix D_{exp} using the assumption that all the nucleotide positions in the HRE sequences are completely independent. The elements of the expected matrix are thus calculated as follows:

$$D_{exp}(bd, j) = v_{j,b} \times v_{j+1,d}, \quad (3.6)$$

where bd , $b, d \in \{A, C, G, T, gap\}$ is a di-nucleotide (e.g. CpG), and $v_{j,b}$, $j = 1, \dots, m-1$, $b \in \{A, C, G, T, gap\}$, are the elements of the mono-nucleotide PFM. That is, the relative frequency (i.e. probability) of occurrence of the di-nucleotide in the position j is equal to the product of the relative frequencies of its first nucleotide at the position j and the second nucleotide at the position $j+1$. Fig. 3.5 illustrates the idea of computing the elements of the expected matrix by the example of $D_{exp}(\text{CpG}, 1)$.

- Second, we compute the actual di-nucleotide frequency matrix D using the collection of the aligned HRE sequences.

- Third, we compare the expected di-nucleotide PFM D_{exp} with D using the chi-square homogeneity test. If the expected matrix calculated with the assumption of

nucleotide independence differs from the actual matrix significantly, we can conclude that there is some di-nucleotide preservation or elimination in the functional HRE sequences.

Fig. 3.5 Computing $D_{exp}(\text{CpG},1)$ using an example PFM.

If the expected and the actual di-nucleotide HRE PFMs are statistically different, we proceed the order justification for the PWM-based HRE modeling to higher values in a similar way. In particular, for determination of the tri-nucleotide preservation in HREs, we calculate the three *expected* tri-nucleotide frequency matrices using the similar assumption of nucleotide positional independence as follows:

$$T_{exp}^1(bdf,j) = v_{j,b} \times D(df,j+1) \quad (3.7)$$

$$T_{exp}^2(bdf,j) = D(bd,j) \times v_{j+2,f} \quad (3.8)$$

$$T_{exp}^3(bdf,j) = v_{j,b} \times v_{j+1,d} \times v_{j+2,f} \quad (3.9)$$

where bdf , $b, d, f \in \{A, C, G, T, gap\}$, is a tri-nucleotide, and $D(i,j)$, $j = 1, \dots, m-2$, is the element of the actual di-nucleotide PFM. The underlying idea is that the probability of simultaneous independent outcomes is equal to the product of the probabilities of these outcomes. The outcome here is either one of the four mono-nucleotides in a given position, or one of sixteen di-nucleotides, or one of sixty four tri-nucleotides. Comparing the expected matrices with the actual tri-nucleotide PFM T , we can determine if there is a need of using tri-nucleotide patterns for HRE modeling.

If there exists the confirmed di-nucleotide preservation within the HRE pattern then it is unlikely that the T_{exp}^3 will be similar to T , as their difference is already

implied at the di-nucleotide level. However, if in fact there is no tri-nucleotide preservation, then the expected matrices T^1_{exp} and T^2_{exp} should demonstrate high similarity to the actual matrix T .

Table 3.3 Results of order justification for the PWM model.

a) 2 nd order model			b) 3 rd order model					
X_n^2	Probability of X_n^2	T^1_{exp} vs. T		T^2_{exp} vs. T		T^3_{exp} vs. T		
		X_n^2	Prob. of X_n^2	X_n^2	Prob. of X_n^2	X_n^2	Prob. of X_n^2	
1	44.0	3.2E-06	108.3	1	96.8	1	132.3	1
2	33.4	2.2E-04	70.4	1	52.1	1	83.3	1
3	14.6	0.146	76.5	1	77.6	1	90.3	1
4	20.4	0.025	76.3	1	101.8	1	119.6	1
5	51.6	1.3E-07	125.8	1	97.2	1	137.3	1
6	23.2	0.010	91.9	1	136.6	1	147.2	1
7	69.1	6.4E-11	134.8	1	126.5	1	187.1	0.99
8	58.0	8.4E-09	110.1	1	117.7	1	163.4	0.99
9	62.0	1.4E-09	113.4	1	124.9	1	174.7	0.99
10	74.5	5.8E-12	132.0	1	104.7	1	170.8	0.99
11	46.9	9.6E-07	88.7	1	56.8	1	100.1	1
12	15.9	0.108	24.6	1	9.5	1	24.8	1
13	0.6	1.00	3.8	1	3.3	1	3.9	1
14	0.06	1.00	0.113	1	0.3	1	0.37	1
15	0.2	1.00	3.1	1	12.3	1	12.5	1
16	6.6	0.761	38.6	1	46.3	1	51.3	1
17	19.5	0.033	27.4	1	43.1	1	47.8	1
18	22.7	0.011	62.1	1	83.6	1	110.8	1
19	42.1	7.2E-06	125.9	1	133.7	1	162.1	0.99
20	50.4	2.2E-07						

The method for verifying the preservation of (N)bp-long oligo-nucleotides can also be designed in a similar way. The number of *expected* oligo-nucleotide matrices, and the number of matrix-to-matrix comparisons, will be $2^{N-1}-1$.

After comparison of the expected di-nucleotide PFM D_{exp} with the actual di-nucleotide PFM D , we found out that it would be indeed informative to include di-nucleotide pattern analysis. Table 3.3a shows the values of the homogeneity statistical

function X_n^2 for the nucleotide positions within the HRE sequence together with its probabilities. Indeed, most of the p-values in the table are lower than a tolerant significance level resulting in the statistically significant difference between the expected and actual di-nucleotide PFMs. Hence, the 2nd order PWM model for HRE recognition is of necessity. Additionally, Table 3.3b shows the values of the statistical function X_n^2 for the HRE positions obtained using the expected and the actual tri-nucleotide PFMs. As opposed to the case of di-nucleotides, the preservation of tri-nucleotides can hardly be observed. The fact that T_{exp}^3 is slightly less similar to T than the other two expected tri-nucleotide matrices is not surprising since it can be explained by the di-nucleotide preservation within the HRE pattern which has been identified earlier.

Eventually, we conclude that the necessary and sufficient order of the PWM model designed for HRE recognition is 2, and the 3rd order modeling is excluded

3.3.2 The first-order weight matrix model

We computed the PFMs for the three steroid HREs of interest, and then computed the joint PFM and PWM using the entire set of HREs.

The results of the preliminary tests using the collection of experimentally validated HREs demonstrated that a reasonable threshold β for HRE prediction was around the value of 0.85. Then, for the model validation, only the neighboring values (± 0.1) were investigated in details.

We ran series of experiments to test the first-order model that was based on the mono-nucleotide frequencies of HREs. In each experiment, we performed five runs. In each run of the first experiment, we randomly selected 70% of collected HREs for training the model, and the rest 30% were used for testing. In the second experiment, equal division of the HRE collection into training and testing sets was used.

The results of the experiments are summarized in Table 3.4. In the table, RE value is the random expectation, which is the length of neutral DNA sequence with on average 1 prediction. It is measured using 1Mb of DNA randomly generated with assumption of equal probabilities for all four nucleotide bases.

The ROC curve (receiver operating characteristic, which is a graphical plot of the sensitivity vs. 1-specificity) for the first-order PWM model used for HRE prediction was also generated (Fig. 3.6), and the AUC (area under curve) value was found to be

as high as 0.953.

Table 3.4 HRE prediction accuracy for the first-order PWM model.

β	Sensitivity. %										RE, bp
	Train/Test: 70/30					Train/Test: 50/50					
	1	2	3	4	5	1	2	3	4	5	
0.77	94.9	92.6	96.6	96.0	94.9	95.2	91.1	92.9	95.2	94.6	1128
0.78	94.3	92.0	96.6	96.0	94.3	95.2	91.1	91.7	95.2	94.6	1149
0.79	93.1	91.4	95.5	94.3	92.6	94.6	88.2	91.1	93.5	92.9	1169
0.80	92.6	90.2	94.9	94.3	90.8	94.0	87.7	90.6	93.5	92.3	1199
0.81	90.2	86.7	92.6	93.7	89.7	92.9	85.3	89.4	88.2	90.0	1238
0.82	89.7	86.2	92.6	93.1	89.1	91.1	85.3	88.8	87.7	89.4	1312
0.83	89.1	84.4	90.2	92.6	88.5	89.4	84.2	86.5	87.1	87.1	1384
0.84	86.2	80.3	86.7	88.5	85.0	87.1	81.3	80.7	81.3	84.2	1416
0.85	84.4	79.2	84.4	87.9	83.8	84.2	80.1	77.8	79.5	83.0	1434
0.86	79.2	72.8	78.6	84.4	78.6	78.3	76.6	73.1	74.3	76.0	1526
0.87	76.3	71.0	76.9	81.5	75.7	77.2	74.9	69.0	71.4	73.7	1624
0.88	72.8	67.6	71.6	78.6	72.8	72.0	72.5	66.7	67.3	70.8	1909
0.89	69.9	65.8	69.3	75.7	69.3	67.9	69.6	65.0	65.6	70.8	2428
0.90	67.0	64.7	65.8	73.4	67.0	64.4	67.3	63.8	63.8	68.5	2656
0.91	63.5	61.7	64.1	69.3	64.7	62.1	65.6	61.5	59.7	65.6	3489
0.92	59.4	59.4	61.2	65.8	63.5	57.4	62.7	58.6	57.4	61.5	3595
0.93	57.1	57.1	57.1	62.9	59.4	55.1	60.3	56.8	55.7	58.0	4323
0.94	49.0	47.8	51.9	53.6	50.7	46.4	52.8	47.0	51.0	47.0	5630
0.95	44.3	44.9	47.2	47.2	44.9	38.2	48.1	41.1	44.6	42.9	7031

Notation: β - PWM threshold, RE – random expectation value.

Fig. 3.6 The ROC curve for HRE prediction by the first-order PWM model.

3.3.3 The second-order model

In order to make the PWM-based modeling of HREs more precise (and possibly more accurate), we also implemented the second-order PWM model for HRE recognition. However, unlike the case of the first-order model where the similarity of the input is a simple convolution of the mono-nucleotide PWM with the aligned sequence, the similarity calculations using the second-order model requires prior preparation which is performed as follows. The input sequence is at first transformed using the di-nucleotide alphabet. Then, for the transformed sequence, its matrix similarity is calculated similarly to that for the mono-nucleotide PWM, but the matrix in use is the di-nucleotide PWM. Normalization coefficient for C_i calculation (refer to equation (3.3)) also should be updated. As the number of different di-nucleotides (and the corresponding number of letters in the newly implemented alphabet) is 16 or 20 or 25, depending on whether the alignment gap is applicable or not, it is necessary to replace the normalization coefficient by the alphabet size.

Table 3.5 HRE prediction accuracy for the second-order PWM model.

β	Sensitivity. %										RE, bp
	Train-test: 70-30					Train-test: 50-50					
	1	2	3	4	5	1	2	3	4	5	
0.67	96.2	97.9	97.9	97.9	94.5	98.4	88.0	89.8	91.5	94.9	623
0.68	96.2	96.2	92.8	97.9	91.0	98.4	88.0	89.8	89.8	93.2	789
0.69	94.5	94.5	91.0	96.2	91.0	98.4	86.3	88.0	89.8	91.5	811
0.70	91.0	91.0	91.0	96.2	91.0	98.4	86.3	88.0	89.8	91.5	923
0.71	91.0	87.6	84.1	91.0	89.3	98.4	86.3	86.3	89.8	88.0	987
0.72	91.0	85.9	79.0	87.6	85.9	94.9	82.9	86.3	82.9	84.6	1025
0.73	89.3	82.4	72.1	80.7	80.7	93.2	79.4	82.9	81.1	82.9	1114
0.74	89.3	82.4	68.6	79.0	77.2	88.0	72.5	72.5	76.0	81.1	1221
0.75	84.5	82.4	63.4	75.5	75.5	84.6	69.1	72.5	76.0	77.7	1369
0.76	79.8	82.4	60.0	72.1	70.3	81.1	67.3	72.5	67.3	72.5	1433
0.77	73.6	80.7	58.3	70.3	68.6	81.1	65.6	70.8	67.3	70.8	1711
0.78	68.6	73.8	53.1	63.4	66.9	77.7	63.9	67.3	63.9	67.3	1810
0.79	61.7	72.1	49.7	56.6	60.0	77.7	60.4	65.6	57.0	65.6	2456
0.80	58.3	70.3	47.9	53.1	60.0	74.2	53.6	60.4	53.6	62.2	3289
0.81	56.6	61.7	41.0	39.3	54.8	63.9	51.8	57.0	51.8	58.7	3395
0.82	56.6	54.8	37.6	35.9	47.9	43.2	50.1	50.1	41.5	50.1	3798
0.83	46.2	49.7	30.7	27.2	35.9	39.8	46.7	44.9	34.6	48.4	4123
0.84	42.8	49.7	29.0	27.2	32.4	38.0	43.2	43.2	31.1	41.5	5249

Notation: α - PWM threshold, RE – random expectation value.

To measure the sensitivity of the second-order model, we used the same experimental setup as that for the mono-nucleotide PWM described in Section 3.2.2. However, for the second-order model we had the typical matrix similarity coefficients for the experimentally validated HRE sequences to be around the value of 0.75.

The results of the HRE prediction using the second-order weight matrix model are summarized in Table 3.5. Additionally, Fig. 3.7 shows the ROC curve with the sensitivity vs. specificity balance for the di-nucleotide PWM-based HRE prediction. The predicted AUC value was found to be as high as 0.941.

Fig. 3.7 The ROC curve for the second order PWM-based prediction of HREs.

To conclude the chapter, PWM models can be successfully used for HRE recognition when learned using an exhaustive experimental knowledge. Correlation analysis allows revealing the necessary and sufficient complexity of motif modeling. For the case study of hormone response elements, second-order weight matrix allows to involve the sequence properties that cannot be manifested based on solely mono-nucleotide frequencies.

Chapter 4

Construction of Markov Model for Description of State Transitions in HRE

In the previous chapter, we presented the PWM-based position frequency model and evaluated its applicability to recognition of symmetrically structured DNA motifs by the example of HREs. While the statistical characteristics of each nucleotide position modeled by the PWM are undoubtedly important for pattern recognition and also allow reconstructing an easily interpretable HRE motif profile, we believe that statistic features between the nucleotides, or inter-nucleotide transition patterns, may provide a new dimension in modeling of the sequence motifs. Hence, in this chapter, we study a method for recognition of HREs which is based on the profile Markov model specifically designed according to the known dimeric structure of the HRE consensus.

A cascade Markov model with a specific state transition matrix for each component of the complex HRE structure is proposed. In Section 4.1, we introduce the concept of the cascade profile Markov model for the HRE pattern. Five components of the Markov model are designed and trained in correspondence with the known HRE structure. To apply the new Markov model, in Section 4.2, we propose a combined position-transition model for the HRE pattern. The combined model includes the two PWM-based models from the previous chapter and the cascade Markov model for enhanced HRE prediction. In Section 4.3, we conduct analysis and comparison of the HRE prediction results with reference to the proposed statistic models, namely, the cascade Markov model, the PWM models, and the combined position-transition model.

4.1 A Cascade Markov Model for HRE Recognition

A Markov model is an extension of a finite state machine with the Markov property [137]. Markov models are widely used for different problems of bioinformatics. In particular, the Markov model can be used to represent the

probability of transitions between the adjacent nucleotides in a DNA sequence.

A basic Markov model of a DNA sequence is shown in Fig. 4.1. Each of the four possible nucleotide bases is represented by the respective state of the Markov model (A, G, C, and T), and the two additional states (B and E) stand for the beginning and ending of the sequence, if applicable. An arc represents a possible transition from one state to another (including itself). With reference to the Markov property, an input DNA sequence can be considered as a chain of Markov state transitions, and the probability of the sequence sharing the same transition pattern as the specifically trained Markov model is determined by the probabilities of its corresponding state transitions.

Fig. 4.1 State-transition diagram of a basic Markov model for a DNA sequence

However, the above naïve Markov model does not associate the transition probabilities with the position of the nucleotides in the sequence. For representation of complex motif patterns like HREs, it is necessary to define a more sophisticated model in order to incorporate the targeted sequence pattern into calculations of the Markov probabilities. With reference to the suggested HRE structure (Fig. 2.3), we propose a five-stage model in which the two HRE half-sites, the internal spacer, and the two flanking regions are considered as five *components* of the *cascade* model. Each component model has its own transition parameters, while the entire multi-stage model computes the transition probabilities between them.

4.1.1 Component Markov model

Each constituent of the HRE pattern, namely the half-site, the spacer, and the flanking region, is implemented in a form of a simple Markov chain similar to the one

shown in Fig. 4.1, and these models are then considered as component elements of the complex five-stage HRE model. Below we describe the sequence representation used in each of the component Markov models in details.

Let $Q = [s_1 \dots s_N]$ be a finite set of states for the putative chain. For representation of the HRE pattern, N ranges from 4 to 6 depending on whether the beginning state B and/or the ending state E are considered. In particular, the HRE flanking regions are modeled with either state B or E, while the internal component models contain neither.

In general, let $s^t \in Q$ be the state of the Markov model observed at time t . With reference to the first-order Markov property, the probability of the model to be in the state s^t at time t given that at times $t-1, t-2, \dots, 0$, the model was in states $s^{t-1}, s^{t-2} \dots s^0$ is as follows:

$$P(t, s^t | s^{t-1}, s^{t-2} \dots s^0) = P(t, s^t | s^{t-1}). \quad (4.1)$$

Then, the probability of the complete way $(s^t, s^{t-1}, s^{t-2} \dots s^1, s^0)$ passed by the model from the beginning of its operation till t is as follows:

$$P(s^t, s^{t-1}, s^{t-2} \dots s^1, s^0) = P(s^t | s^{t-1}) P(s^{t-1} | s^{t-2}) \dots P(s^1 | s^0) P(s^0), \quad (4.2)$$

where $P(s^0)$ is the initial probability of the model to be in state s^0 at time 0. The probabilities of transitions $P(s^t = i | s^{t-1} = j)$ are thus determined using the *state transition matrix* $\mathbf{P} = [p_{ij}]_{N \times N}$ where p_{ij} is the probability of the transition from state $i \in Q$ to state $j \in Q$ [138]. The state transition matrix must be stochastic, i.e. its elements should be non-negative, not exceeding 1, and the elements of each row of this matrix are to be summed to 1.

For the particular case of the HRE DNA sequences represented by the Markov chain [131], every nucleotide of the sequence corresponds to the state of the model (refer to Fig. 4.1), and the time flow for the Markov model is reflected by the propagation along the motif sequence. Let the HRE sequence of length m be denoted as $S = (s_1 \dots s_m)$, $s_i \in \{A, C, G, T\}$, $i = 1, \dots, m$. Then the Markov probability of that sequence is calculated as follows:

$$P(S) = P(s_m | s_{m-1}) \cdot P(s_{m-1} | s_{m-2}) \cdot P(s_2 | s_1) \cdot P(s_1) = P(s_1) \prod_{i=2}^m a_{s_{i-1}, s_i} \quad (4.3)$$

where $a_{k,l} = P(s_i = l \mid s_{i-1} = k)$, $k, l \in \{A, C, G, T\}$, is the probability of the nucleotide l to be followed by the nucleotide k . The method requires the parameters for the initial probability $P(s_1)$ and the transition probabilities a_{ij} to be defined. The states B and E can be added to present the model in a uniform style where the initial probability $P(s_1)$ corresponds to the transition probability from state B to state s_1 .

4.1.2 Parameter estimation

Both the component and the cascade Markov models require transition probabilities to be defined before being used for HRE prediction. The process of defining these probabilities is referred to as Markov training.

Let a set $D = \{S = (s_1, \dots, s_m) \mid s_i \in \Omega_{DNA}, i = 1, \dots, m\}$, where Ω_{DNA} is the DNA alphabet, be the training set of HRE sequences used for determining the empirical transition probabilities. The goal of the parameter estimation is to infer the transition probabilities for a pre-defined Markov model M (either component or cascade) using the training set D . A common statistical approach for model training, which allows to maximize the likelihood of representation the HRE dataset by the resulting model, is known as the maximum likelihood parameter estimation (ML). The maximum likelihood model M^{ML} for HRE recognition is defined as

$$M^{ML} = \arg \max_M P(D \mid M) \quad (4.4)$$

As the HRE sequences from the dataset D are assumed to be independent, the likelihood of the model M with reference to the HRE training set D is calculated as follows:

$$P(D \mid M) = \prod_{S \in D} P(S \mid M) \quad (4.5)$$

Logarithmic transformation does not affect the location of the maximum for the likelihood function, so we can take the logarithm of the both sides of the previous equation:

$$\log P(D \mid M) = \sum_{S \in D} \log P(S \mid M) = \sum_{S \in D} \sum_{i=1}^l \log P_i(s_i \mid s_{i-1}, \dots, s_1) =$$

$$= \sum_{s \in D} \sum_{i=1}^l \log P_i(s_i | s_{i-1}). \quad (4.6)$$

The last transformation is made due to the assumption of the Markov property for HRE modeling.

Under the maximum likelihood estimation which provides the optimum of the above likelihood function with reference to the HRE training set, after calculating all partial derivatives of the above equation we obtain that the maximum likelihood probabilities are equal to the empirical probabilities, or frequencies, of the corresponding transitions [131]:

$$a_{s,t} = \frac{n_{s,t}}{\sum_{t' \in \Sigma} n_{s,t'}} \quad (4.7)$$

where $a_{s,t}$ denotes the transition probability from the state s to the state t , and $n_{s,t}$ is the number of state transitions from the state s to the state t observed using the HRE training set. That is, as implied by the maximum likelihood approach in its general form, the probability of the transition is associated with its observed frequency.

To compute transition probabilities for the component Markov models, the di-nucleotide frequencies are first collected from the HRE training set. Then, these frequencies are converted into transition (i.e. conditional) frequencies and are associated with the corresponding probabilities. The transition probabilities for inter-model transitions in the cascade model are calculated using the maximum likelihood approach as well.

4.1.3 Representation of the HRE complex structure with a cascade Markov model

The binding site for the steroid hormone receptor consists of three domains: the two half-sites, and a spacer. The defined positions of the HRE half-sites are highly conserved due to the presence of contact points typical for nuclear receptor response elements. In our model, we also consider flanking regions, because they have previously been reported to be involved into the process of protein-DNA interaction by Haelens et al. [126]. Therefore, the number of component domains to be included into

the model is five.

Each of the component domains is expected to have its own properties (i.e. internal transition probabilities), so the component Markov models for the corresponding domains have to be examined and trained separately. It is also necessary to consider that the two half-sites of the HRE sequence have different degrees of conservation. In particular, the right half-site has a highly rigid structure, while the left half-size allows for relatively tolerated mismatches in most of its nucleotide positions.

Following the above, we propose the cascade Markov model (shown in Fig. 4.2) for representation of steroid HREs. Its multi-component structure satisfies the described conditions: different domains are considered independently, and the transition probabilities between them are determined. In particular, the transition probabilities for the first component model, which corresponds to the states 1 through 4, are determined using the left 3bp-long flanking regions, the second component model (states 5-8) represents the left HRE half-site, and so on. The fact that the right half-site is highly conserved is reflected in the structure of the fourth component model, which corresponds to the states 13-31. The transition probabilities in that part of the cascade model are explicitly position-dependent, unlike the transition probabilities in the other parts of the sequence. Based upon the collected HRE PFMs, where the GpT di-nucleotide is observed in 99% of sequences of the right HRE half-sites, we decide that the presence of that GpT is essential for protein-DNA binding. Thus, if that di-nucleotide is absent at the specified position modeled by the states 18 to 21, the sequence under testing cannot not be an HRE (transition to the state 19).

Given the length of each constituent part of the HRE, i.e. the flanking regions are of length 3, etc., we define the number of possible transitions inside each of the component Markov models. As an illustration, there are two transitions allowed in the flanking regions models as shown in Fig. 4.3, so that three states are visited and the 3bp-long sequence is returned.

There are some variations in lengths of the training sequences as not all of them are annotated with flanking regions in literature. Hence, normalization procedure for the Markov probability value is used – logarithm of the probability is divided by the sequence length. Also a prior distribution, if the sequence starting position is different from B, is taken from the HRE PFM; otherwise, it is considered as uniform.

After the sequence is processed by the model, its Markov probability is stored into the HRE database and is subject to a threshold for decision making.

Fig. 4.2 The five-stage cascade Markov model for HRE recognition.

Similarly to the weight matrix model, for the screening of long DNA sequences, a sliding 21bp-long window is used. In that case, the application returns the list of recognized HREs with their relative positions within the sequence. However, it is necessary to note that the proposed architecture of the cascade Markov model is specifically designed to be used as a part of the combined position-transition model, that is, for screening lists of the putative HREs returned by less complex methods such as PWMs. In order to be used as a standalone solution for analysis of long genomic regions, this Markov model should be accelerated by a systolic array or a similar streaming technology for reducing communication overheads.

4.2 The Combined Position-Transition Model and Tiger HRE Finder

While the mono- and di-nucleotide PWMs can be used to extract the statistic patterns with reference to the nucleotide composition of the motif sequence, the cascade Markov model exploits a different approach and provides a versatile method for the transition pattern modeling. It is reasonable to believe (and this strategy is already widely used for design of multiple-feature frameworks and highly specific ensemble models [16]) that if two or more pattern recognition methods are designed for considering different properties of the underlying object, then we may outperform each single method operating solely by aggregating their predictions. This makes sense as the independent and distinct recognition methods tend to make errors under different conditions and misclassify different objects, thus providing a more accurate description of the pattern of interest when combined.

In our implementation, the combined recognition of the HRE pattern using a unanimous voting scheme (Fig. 4.3) operates as follows. An HRE is considered as recognized and stored into the database only if the first- and second-order PWM models support the prediction returned by the cascade Markov model. Presence of three thresholds provides us with a very flexible tuning mechanism for enhanced HRE prediction.

Additionally, we propose the tool for HRE recognition, named Tiger HRE Finder, which is now available online at <http://birc.ntu.edu.sg/~pmaria/> (Fig. 4.4). The web version of the model was written in Perl for UNIX/Linux. A user can submit a sequence up to 5kbp length at a time in any conventional DNA format (plain text, EMBL, GCG, FASTA, Genbank, IG), select a method for HRE prediction, and define

a threshold for the method. Table of thresholds with corresponding sensitivities and prediction rates (note that they are different for the PWMs and the Markov model) is given in the Supplemental Info on the web site so as to define the prediction parameters more specifically.

Fig. 4.3 The combined position-transition model for HRE recognition.

The screenshot shows the web interface of the Tiger SHRE Finder 1.0 tool. The browser window title is 'Tiger SHRE Finder 1.0 - Mozilla Firefox'. The address bar shows 'http://birc.ntu.edu.sg/~pmaria/'. The main content area has a green header with a tiger logo and the text 'TIGER SHRE Finder 1.0'. Below this, it says 'Recognition of Steroid Hormone Response Elements in DNA sequences for androgen, glucocorticoid and progesterone hormones'. A 'Table of Contents' section includes links for 'Home' and 'Supplemental Data'. The 'Threshold' is set to '0.85' with a note: '(correspondent accuracy of prediction - see table)'. The 'Prediction Method' is set to 'PWM' (radio button selected) with a note: 'Please note that reasonable threshold values for these methods are different: for PWM ~0.8-0.9, for HMM ~0.30-0.35'. The 'Sequence' input area contains a long DNA sequence. Below the sequence, there is a checkbox for 'Include complementary DNA strand' and a 'Submit' button. A 'Clear' button is also present. The browser status bar at the bottom shows 'Done', '0.471s', and 'AdBlock'.

Fig. 4.4 Web interface of the Tiger HRE Finder tool.

4.3 Experimental Results

Similar to the position frequency-based weight matrix models, the Markov transition model requires a trade-off between prediction sensitivity and specificity. To evaluate the sensitivity of our new model we used the same experimental setup as for the mono- and di-nucleotide position weight matrix models from Chapter 3 with different proportions of the collected dataset of HREs used for training and testing. The typical Markov probability values for functional HREs were found to be around 0.35. The accuracy of HRE prediction by the five-stage Markov model is summed up in the Table 4.1.

Table 4.1 Results of HRE prediction by the five-stage cascade Markov model.

β	Sensitivity. %										RE, bp ⁻¹
	Train/Test: 70/30					Train/Test: 50/50					
	1	2	3	4	5	1	2	3	4	5	
0.26	1	1	1	1	99.3	1	1	99.3	1	1	64
0.27	98.8	98.5	99.3	98.8	97.8	97.8	99.3	99.2	95.7	99.5	90
0.28	97.7	97.1	93.9	97.8	97.1	97.4	94.3	98.8	94.3	98.1	130
0.29	96.0	94.3	92.9	95.0	95.0	96.4	93.6	97.8	93.2	95.0	190
0.30	95.2	93.9	92.5	94.6	94.6	93.9	92.2	96.0	91.8	94.3	295
0.31	94.8	92.5	91.1	94.3	93.9	93.2	91.8	95.3	91.1	92.2	478
0.32	94.4	91.8	89.7	93.2	93.2	92.2	90.8	94.3	90.1	89.0	641
0.33	88.3	89.7	87.3	90.8	91.1	85	84.6	90.8	86.9	87.6	855
0.34	83.1	85.9	81	78.2	84.4	73.7	75.8	81.2	74.2	70.3	1217
0.35	73.5	75.5	74.7	73.6	68.7	66.3	62.8	75.6	64.6	66.7	2049
0.36	64.8	64.5	61.7	67.7	63.8	61.8	57.2	61.1	57.6	62.5	3521
0.37	60.6	61.0	58.3	63.8	60.3	58.3	53.0	57.6	52.3	59.3	6452
0.38	50.1	50.6	48.8	52.7	52.0	46.8	42.2	48.9	39.4	48.2	19037
0.39	32.4	30.1	30.1	32.2	31.1	27.0	22.4	26.6	20.3	28.7	71429
0.40	23.4	22.0	22.0	24.1	23.4	18.6	14.4	17.9	14.4	18.2	142857

Notation: β - the Markov probability threshold, RE – the random expectation.

In the Table, the results of ten independent accuracy tests are given. First, five test were performed with 70% of the HRE collection used for training and the rest used for testing. Second, the training/testing split into half and half was used, with the sequences for training being randomly selected from the HRE database without

replacement. For the five-stage Markov model, we achieved 90% of sensitivity with the prediction rate of 1:755bp (the threshold for the normalized Markov probability of 0.325), and the level of prediction rate of 1:6.5kb associated with 40% of correctly predicted HREs (the threshold value of 0.385). The AUC value for the cascade Markov model is 0.924, which is lower than those for the position weight matrix models, but still in the range of excellent prediction accuracies typical for the problem of TFBS recognition [35].

Using the combined position-transition HRE recognition model, we expected to eliminate a certain amount of false positives. The prediction accuracy for the combined model was tracked in comparison with each of the methods involved. Figure 4.5 shows the ROC curves for different single- and multiple-feature HRE prediction frameworks. The ROC curve for the first-order PWM model is labeled with PWM1. The ROC curve for the second-order weight matrix model based on the di-nucleotide composition is labeled with PWM2. The curve for the two weight matrix models used together is labeled with “PWMs”. Label MM corresponds to the multi-stage cascade Markov model described in this Chapter. Finally, the label “Tiger” denotes the ROC curve for HRE prediction by the combination of all the three prediction models implemented as a unanimous voting scheme.

Fig. 4.5 The ROC curves for the combined prediction of HREs in comparison with the first- and second-order PWM models and the five-stage cascade Markov model.

For the multi-stage predictors, such as “PWMs” or “Tiger”, plotting of ROC curves requires tuning of two or three threshold values simultaneously for balancing the prediction rate and sensitivity. As an approximate solution, we change one threshold parameter at a time by a small value, and estimate average prediction specificity for each sensitivity level in a series of experiments.

Sensitivity and random expectation for ensemble methods are functions of two or more variables and can be tuned in a respective multi-dimensional space. Additionally, exploiting a multiple-feature framework with three prediction steps, we eliminate a number of the false positives while keeping the number of true positives. For the problem of HRE prediction, we set the following values for prediction thresholds: for PWM1, the prediction threshold is 0.85; for PWM2, the prediction threshold is 0.73; for MM, the prediction threshold is 0.34. Using these threshold values, we received a combination of 73% for the HRE prediction sensitivity and the random expectation of 1 hit per 4.74kb. The resulting AUC is 0.968, in comparison with that for the five-stage Markov model which is 0.924, and for the first- and second-order PWMs which are 0.953 and 0.941 respectively. When the two position frequency models PWM1 and PWM2 are used together but without involving the transition pattern, a combination of 76% sensitivity and the prediction rate of 1 hit per 3.18 kb of neutral DNA can be declared as a typical HRE prediction accuracy (the respective AUC is 0.963).

In conclusion to the chapter, Markov model is a powerful tool for sequence modeling. For the problem of modeling symmetrically structured HREs, its specifically adapted design allows us to achieve a good accuracy of prediction when used as a component of combined position-transition statistic model. Moreover, the complex structure of the designed Markov model grants one a plenty of ways for further improvement, in particular, via the scheme restructuring and order management. Thus, the proposed model is ideal for implementation in a form of a high-performance solution which is especially important for the computationally expensive bioinformatics problems such as sequence analysis.

Chapter 5

Empowering Accurate HRE Modeling with FPGA Implementation

In the previous chapters, we described the two statistic approaches for HRE recognition based on single and multiple nucleotide composition of the HRE motifs, namely the position weight matrix and the profile Markov model. When used in combination, these approaches outperformed each of them operated alone. However, in addition to its predictive capabilities, each algorithm, either a naïve or an ensemble, is characterized by its complexity.

In the software implementation of bioinformatics algorithms, we sometimes limit the modeling accuracy to avoid the prohibitively long execution time when dealing with huge amounts of sequence data. In this chapter, we test the applicability of the Field-Programmable Gate Array (FPGA) technology of application-specific hardware acceleration to the problem of genome-wide recognition of symmetrically structured DNA motifs, by the example of the specifically designed cascade Markov model. Using the FPGA, we develop a hardware-accelerated architecture to exploit the modeling capabilities of the proposed Markov model in full.

The developed cascade Markov model consists of several component models, and though they are represented sequentially, these models can be processed in parallel for each given sequence. Thus, by implementing the cascade model on FPGA, we can achieve significant acceleration with fine-grained parallelization and optimization of logic interconnections. The latter is an advantage of the FPGA technology in comparison with general purpose processors.

In this chapter, we describe the FPGA implementation of the cascade Markov model designed for recognition of HREs. In Section 5.1, we introduce the concept of parallel processing of the component Markov models, and in Section 5.2, we describe their implementation together with interconnections between the component models. In Section 5.3, we discuss the computational performance of both software and hardware HRE prediction solutions, and summarize the models described in the Part II of this thesis in Section 5.4.

5.1 Fine-Grained Parallelization with FPGA

An FPGA chip basically consists of three types of blocks: input/output blocks (IOB), configurable logic blocks (CLB) and block RAM (Fig. 5.1). The CLB can implement different digital functions, programmable interconnects, and routing resources to allow different blocks to be connected together. The IOB provides the interface to the outside world, for example, to the awaiting front-end application when the FPGA chip is used as a co-processor. The RAM structure blocks could be configured as RAM to facilitate memory type of logic when the algorithm calls for one. For implementations and tests of the cascade Markov model on FPGA, we used the ADM-XRC-4 PCI board with the Virtex-4 FPGA chip which contained 135,168 logic cells and 5,184Kbit of embedded RAM.

The design flow for the FPGA implementation of the cascade Markov model is summarized in Figure 5.2. In particular, we simulate the design at multiple stages throughout the design process. Initially, the register-transfer level description (RTL) developed using Verilog HDL is simulated by creating the test benches. Then, after the synthesis engine maps the design to a netlist, the netlist is translated to a gate level description where the simulation is repeated to validate the synthesis results. Finally, the design is laid out in the FPGA chip. At that stage, propagation delays are added and the simulation is run again with these values back-annotated onto the netlist. The configured FPGA chip is used as a co-processor for the hybrid CPU-FPGA computing system.

Fig. 5.1 FPGA structure for implementation of the cascade Markov model.

Fig. 5.2 Design flow for development of the parallel architecture for implementation of the cascade Markov model on FPGA.

We used Verilog HDL language for the FPGA design, and the ADM SDK library provided by the board manufacturer for the CPU-to-FPGA communication. The SDK library contained API functions essential for driving the FPGA chip by the host PC. The front-end application was developed using Microsoft Visual Studio 2005.

The hybrid CPU-FPGA system operates as follows. First, the C++ application reads DNA sequences from a text file or memory, converts the letters of the DNA alphabet into binary numbers using two bits per nucleotide positions and 21 nucleotide

positions per input, and sends the 42bit-long input vector to the FPGA board. It also obtains the output Markov probability from the board, and proceeds it to the decision making scheme for HRE prediction. Markov transition probabilities are calculated using the maximum likelihood approach (Section 4.1.2), and stored in the FPGA memory as 32 bit unsigned fixed point numbers.

The input to the FPGA-implemented Markov model is a DNA sequence. It is then split into five partially overlapping subsequences, each of which is processed by the corresponding component Markov model as shown in Fig. 5.3. In the figure, low case letters “b” and “e” represent beginning and end of the input sequence.

Fig. 5.3 Schematic diagram of the cascade Markov model implemented on FPGA.

The cascade Markov model implemented on FPGA consists of seven main units: the five units for the component Markov models used for processing of the two half-sites of the HRE, two flanking regions, and the spacer between the half-sites; the memory unit, which stores the transition probabilities for each component model; and the merging unit which receives the resulting Markov probabilities from the five component model units and returns the overall Markov probability for the input DNA sequence.

By involvement of the overlapping areas instead of a consequence of component Markov models, we get these models operating in parallel. The overlapping area for splitting of the input sequence is due to the Markov property of the model, so that the current nucleotide state depends on the previous one only. When the first-order Markov model is considered for HRE modeling, one previous nucleotide is merged together with the next component model as its beginning state. For higher-order Markov models, it will be necessary and sufficient just to enlarge the area of overlapping for the sequence splitting.

5.2 Logic Interconnection in the Markov Model

In our implementation, each of the five component Markov models is represented in a form of a Markov processing element (MPE). Two types of Markov processing elements are proposed for the two main types of component models, in particular, the models of the 3bp-long neighboring regions (MPE F, or the Markov processing element for the flanking areas and the spacer), and the 6bp-long HRE half-sites (MPE H, or the Markov processing element for the half-site, shown in Fig. 5.4). All Markov processing elements have their own memory storage units for the corresponding transition probabilities. That is, the physical memory is indeed distributed, since it allows using less area resources in comparison with a common memory block used for the entire Markov model. All the MPEs are connected to the Markov merging element (MME) which is in turn connected to the output of the Markov model FPGA module.

Each MPE has a transition probability detection unit (TPD) for preprocessing of the input sequence (Fig. 5.5). For each input DNA sequence, the TPD returns the memory indexes which are then used for extraction of the corresponding transition probability values from memory.

Fig. 5.4 Dataflow scheme of the Markov processing element MPE H for the cascade Markov model. The MPE F has a similar structure, though uses only two multiplications for the 3bp-long flanking regions and the spacer. The half-site nucleotide sequence is in upper case, while the letter in lower case is its beginning state taken from the preceding element.

Fig. 5.5 The RTL schematic diagram of the TPD unit. The input to the unit is a dinucleotide in the form of four binary numbers, and the output of the unit is the index for the state transition matrix. This index is then used as a pointer for extraction of the corresponding transition probability from memory.

Thus, the transition profile of the input sequence is generated and processed by the on-chip Markov model. Dealing with higher-order Markov models, more preceding elements should be involved, so the transition profile will become (in general, exponentially) larger.

The RTL schemes of the MPE H unit and the entire cascade Markov model are given in Fig. 5.6 and Fig. 5.7, respectively. The RTL schemes were generated from the Verilog HDL source code using the RTLvisionPro tool.

We used the unsigned fixed point notation for representing Markov probabilities. Inside MPEs, 32 bits were used to represent the values of the state transition probabilities, with all 32 bits used for a fractional part; besides, for the merging element, we extended the number width and the size of its fractional part to 64 bits. The reason is that, after a series of tests, 32 fractional bits were found to cause notable underestimation of resulting Markov probabilities for most of the HRE sequences, while 64 bits were confirmed to be enough for the majority of inputs. Indeed, for the length of the HRE sequence of 21 base pairs, the precision of 64 bits for the fractional part of the Markov probability is in general sufficient, though for longer patterns it may be reasonable to involve logarithmic transformations so as to replace multiplication operations by corresponding additions.

In our implementation of the Markov model, arithmetic operations are the standard procedures. In particular, the 36bit×36bit multiplication operates as a finite state machine which regulates the sequence of pairwise 18bit×18bit multiplications accomplished by the two dedicated hardware multipliers, in order not to overload the limited logic gates. Additionally, when 64 bits are used instead of 32 for fractional number representation, four embedded multipliers should be involved instead of two. However, the architecture of frequently used arithmetic operations is always a trade-off between limited resources and computational latency.

The summary of the resulting FPGA implementation of the cascade Markov model is as follows:

Logic elements:	21,396 of 135,168 (16%)
RAM:	160 Kbit of 5,184 Kbit (3%)
I/O pins:	120 of 960 (13%)
DSP slices:	42 of 96 (44%) .

Fig. 5.6 The RTL diagram of the MPE H unit. In the left part, the six squares are the TPD units.

Fig. 5.7 The RTL diagram of the cascade Markov model implemented on FPGA. The five component models implemented as MPEs are shown in the upper left part, and the MME unit is shown in the lower part. The Xilinx ISE 8.1i programming environment (upper right part) was used for Verilog HDL development.

Internal clock frequency of the FPGA chip was set to 100 MHz.

The dataflow during the operation of the cascade Markov model (post-place-and-route simulation) using one input sequence is shown in Fig. 5.8. The signals in the lower part of the figure are those processed by the MPE modules, the signals from the middle section represent the operation of the MME, and in the upper part, the signal `outp` represents the Markov probability of the input DNA sequence to be an HRE as returned by the Markov model unit.

Chapter 5. Empowering HRE Modeling with FPGA

88

Fig. 5.8 Data flow during the operation of the cascade Markov model. The design was simulated using the ModelSim XE/III tool.

5.3 Analysis of Computational Performance and Complexity

For HRE recognition in software, the PWM models, the cascade Markov model and the auxiliary applications for data preprocessing were implemented as C applications for Win32, and compiled using the gcc compiler from the cygwin emulation environment.

The complexity of the mono-nucleotide matrix training and processing is $O(\omega \times L \times N)$ where L is the length of the HRE pattern, ω is the size of the DNA alphabet, and N is the length of the DNA sequence to process or the number of sequences in the training or testing set.

The complexity of the Markov model training is $O(\omega^2 \times L \times n)$ where n is the number of training sequences. In general, we have higher computational complexity for the Markov models than that for the PWMs because the amount of possible state transitions is square to the cardinality of the DNA alphabet. For the proposed five-stage cascade model, however, the computational complexity is lower than it might be on average, because we consider the GpT di-nucleotide in the right half-site of the HRE sequence as an essential contact point for the protein-DNA interaction. Therefore, we can eliminate the candidate sequences which do not have that pattern in the required position. It allows us to gain the average speed-up as much as 16 times solely with the software implementation.

The FPGA speed-up for the cascade Markov model was examined using 1Mb of randomly generated sequences from the DNA alphabet. The list of the testing sequences was preprocessed by the encoding application, and then the sequences were submitted into the board, one at a time, after a handshake signal for the completion of calculations for the previous input was received. In that case, we did not benefit from the possible re-distribution of the data flow, though it was possible to decrease the computational latency even more if the MPE modules started processing the next input while the MME module was calculating the current output.

The results of the tests of computational performance for different implementations of the models can be summarized as follows:

- with the IBM four-processor server (4 CPUs each of 3.17GHz, 3.25GB RAM, Win 2003 Server) it took about 50 seconds to process 1 Megabyte of DNA text by the cascade Markov model, and 60 seconds by the combined

position-transition model;

- by involving the FPGA-based hardware acceleration of the cascade Markov model, it allowed us to screen 1Mb of DNA by that model at 8 seconds of running time.

As it can be noted, the achieved speed-up of 8X is not only due to the parallelization but also due to the exclusive FPGA-specific architecture which exploits the application-specific logic-interconnection design.

5.4 Summary and Discussion

Since TFBS profiles are typically short and degenerate, a signal-to-noise ratio for their *de novo* prediction is usually low, and thus, the TFBSs are intrinsically hard to be recognized precisely. Before we proceed from the statistic position-transition models to more complex pattern recognition methods, we still have to understand the advantages and disadvantages of the models described in this part when applied to the problem of recognition of symmetrically structured DNA motifs.

In this study, for the adapted PWM-based HRE prediction, we found a typical specificity level to be as high as 1 prediction per 3 kilo base pairs, though it was associated with the sensitivity values which were hardly able to exceed the 75th percentile. Developing of the higher-order PWM model is still promising since it allows considering sequence features beyond its single nucleotide composition. At the same time, increasing the order of the PWM model does not cause significant worsening of the calculation time or design complexity.

Unlike PWMs which consider mono- or oligo-nucleotide units of the motif sequence successively, in the novel architecture of the cascade Markov model, several component models reflect the actual multimeric structure of the complex response element. The combination of the two additive statistic approaches, namely, the position-specific PWM and the transition-specific Markov model, results in a significant increase of HRE prediction accuracy as of the 70% sensitivity and the 1:5kbp random prediction rate.

Considering the extensive dataset of HRE sequences which currently has no analogs, our findings are indeed promising. For comparison, the results of the TRANSFAC-based TFBS prediction experiments provided by Rahmann et al. [35]

can be used. In that paper, the authors showed that specificity level higher than 99% could be achieved for only 43 TFBS profiles (i.e. 7%) among 623 used for testing. Some profiles of high interest in practice like nuclear receptor binding sites were not included into that high-quality group. All other profiles resided below the specificity level of 0.99, which is fairly trivial because the corresponding prediction rate is as low as 1 hit per 0.1kb.

To our knowledge, the strategy and development of the cascade Markov models is novel. To make it practically useful, we also studied the applicability of hardware-acceleration technologies because even for the relatively simple statistic models, processing of long genomic sequences in software often results in prohibitively long computing. The novel FPGA architecture was successfully designed and used for HRE prediction, with the calculation speed increased by an order of magnitude. The implemented architecture benefits from both the FPGA fine-grained parallelization and the on-chip application-specific logic interconnections.

The proposed model allows for certain flexibility in hardware design. In particular, the partially parallel structure of the cascade Markov model can be redesigned reflecting the trade-off between latency and area or be mapped onto another chip which may have smaller size and less physical resources. Furthermore, higher-order Markov models have already been required for *in silico* pattern recognition, by the example of genomic signals [89], and undoubtedly will be required in future. With reference to the described multi-stage model, the increase of the Markov model's order will result in a slightly modified memory management and additional logic interconnections for distribution of the input among the processing elements. For example, if we consider a second order Markov model, with reference to Fig. 5.1 we only need to extend the borders of the operation coverage for the processing elements (vertical rectangles along the DNA sequence) and modify the size and control signals for the memory units.

The statistic models of biological sequences harmonize the simplicity and interpretability with good predictive potential. While position-specific models like PWMs allows for little application-specific adaptation, the specifically designed cascade Markov model accompanied with hardware architecture opens the new fields for further research for both bioinformaticians and hardware engineers.

Part III

Adaptable Model for Flexible HRE Pattern

Chapter 6

Dynamically Adaptable Neural Network for HRE Modeling

In Part II of the thesis, we focused on modeling of symmetrically structured DNA motifs using our specifically designed statistic model. The model, when trained using the set of experimentally validated HREs, was shown to work well for HRE recognition. However, it also implied certain limitations. In particular, both weight matrix and Markov model are based on the known consensus of the dimer that has a fixed length for the half-sites and the spacer, and fixed positions of the half-sites within the motif sequence. In nature, however, these assumptions do not always hold, since certain mutations, including insertions and deletions, may only affect binding affinity of the dimeric binding but not completely prevent binding of the transcription factor.

The structure of the HRE motif composed of two 6bp-long half-sites separated by the 3bp-long spacer is kept in most of the real cases. However, in our experiments, the highest sensitivity achieved with this limitation could not exceed 75%, and considering the literature we found numerous evidences of relative tolerance of hormone receptors to more variant HREs. Hence, by exploring a more flexible pattern we expect to obtain a highly sensitive model for HRE prediction. Meanwhile, the specificity can also be improved by using an intelligently adaptable pattern recognition method which reflects the HRE dimeric structure. In this chapter, we propose a dynamically adaptable neural network approach to increase the flexibility of the sequence models, and thus further improve sensitivity of HRE prediction. In its general form, this approach can also be applied to other problems where it is required to recognize TFBSs with symmetric structure in DNA.

An Artificial Neural Network (ANN) is an information processing paradigm that is inspired by biological nervous systems, such as the brain. The neural network system is composed of a large number of highly interconnected processing elements (neurons) working in cooperation to solve specific problems.

Similar to the brain, neural networks configured for specific applications, such as pattern recognition or data classification, learn by examples. The types of applications

where neural networks can be involved include but not limited to [139]: *i*) machine learning: the problems of optimization, classification, associative memory, regression and function mapping; *ii*) cognitive science: modeling higher level reasoning (language, problem solving), modeling lower level reasoning (vision, audition speech recognition, speech generation); *iii*) neurobiology: modeling how the brain works both at the neuron-level and at the higher levels (vision, hearing, etc); *iv*) mathematics: nonparametric statistical analysis and regression; and even *v*) philosophy. The neural networks are particularly used for signal processing, control, robotics, pattern recognition, medicine, business and financial applications, data compression, game playing, etc.

Despite the conceptual simplicity of the approach, using ANNs is not so straightforward, and for proper ANN design, understanding of the underlying theory is essential. Different factors that affect the effectiveness of the approach include:

- Choice of model. This depends on the data representation and the application. Overly complex models tend to exhibit problems with learning.
- Learning algorithm. There are numerous trade-offs for development of learning algorithms. However, almost any algorithm will work well with the correct parameters for training on a particular fixed dataset. At the same time, selecting and tuning an algorithm to train the network for proper operation with unseen data requires a significant amount of experimentation.
- Robustness. If the model, cost function, and learning algorithm are selected appropriately, then the resulting ANN can be extremely robust.

In bioinformatics, ANNs are especially useful for classification and function approximation/mapping problems which have lots of training data available but to which hard and fast rules (similar to those used in expert systems) cannot be easily applied.

The rest of the chapter is organized as follows. In Section 6.1, we present the feed-forward multilayer perceptron designed for modeling of the HRE sequences. In Section 6.2, we describe the Hopfield neural network which it is used for prediction of the dimeric structure of HREs. In Section 6.3, we address the issue of possible instability of recurrent neural networks in the case of biological sequences, as those are usually subject to short mutations. We propose the dynamic-programming adaptation

for more robust dimeric structure prediction. In Section 6.4, we describe a two-phase neural model for HRE prediction, where the first stage is the feed-forward neural network for sequence-based HRE recognition, and the second stage is the Hopfield neural network for dimeric structure prediction. Finally, we summarize and discuss the results in Section 6.5.

6.1 A Feed-Forward Neural Network for Sequence-Based HRE Recognition

Function of a vector of finite dimension on a compact set can be approximated to an arbitrary precision by feed-forward neural networks (FFNNs), provided that enough training data and computing resources for learning are available. In this section, we describe the design of the FFNN for modeling of the set of functional HRE sequences with the aim to use the model for accurate prediction of new HREs.

The set of sequences to be modeled by the FFNN consists of a number of DNA sequences in the 4-letter alphabet $\Omega = \{A, C, G, T\}$. Let $\vec{\theta} = [x_1, \dots, x_L]$, $x_j \in \Omega$ $\forall j=1, \dots, L$ be a DNA sequence of length L , and $\Theta = \{\vec{\theta}_1, \dots, \vec{\theta}_n\}$ be a set of such sequences.

While we are dealing with DNA alphabet, the neural models require numerical representation of the input data. The space of real numbers is one-dimensional, and four nucleotide bases cannot be equidistantly mapped onto it without introducing artifacts to the model. Thus, for numerical representation of the DNA alphabet, we use the *one-hot* encoding scheme which operates as follows. The encoding module Σ is a function on the space of DNA sequences $\Theta - \Sigma: \Theta \rightarrow \Xi_\Theta$, where Ξ_Θ is the space of vectors $\vec{\xi}$ of length $4L$. The elements of this vector are defined as follows:

$$\xi_{4(i-1)+k} = \begin{cases} 1 & \text{if } x_i = \omega_k \\ -1 & \text{otherwise} \end{cases} \quad (6.1)$$

where the DNA sequence under transformation is $\vec{\theta} = [x_1, \dots, x_L]$, $x_j \in \Omega$ $\forall j=1, \dots, L$, and the DNA alphabet elements are $\omega_1 = 'A'$, $\omega_2 = 'C'$, $\omega_3 = 'G'$, $\omega_4 = 'T'$, $\Omega = \{\omega_1, \omega_2, \omega_3, \omega_4\}$. For example, if $x_u = G = \omega_3$, then the u^{th} element of the sequence

is transformed into a 4-vector $(-1, -1, 1, -1)^T$. Thus, the entire DNA sequence $\bar{\theta}$ undergoes a transformation according to that rule: $\bar{\xi} = \Sigma(\bar{\theta})$. In order to use sigmoid activation functions without bungles, we modify the conventional one-hot notation and replace 0 with -1.

Thus, each neural network is forestalled by the encoding module $\Sigma: \Theta \rightarrow \Xi_{\Theta}$. Reverse encoding procedure of the $4L$ -vectors $\hat{\Sigma}: \Xi_{\Theta} \rightarrow \Theta$ is performed using the equation (6.1) as well.

6.1.1 Model definition

We consider a multi-layer perceptron in its basic form that consists of three consecutive layers: the input layer, the hidden layer, and the output layer. Each neuron of each layer is connected to each neuron of the subsequent layer. The connections are represented in the form of weights. A typical example of a feed-forward neural network is shown in Fig. 6.1.

Fig. 6.1 Feed-forward neural network with four input neurons, five hidden neurons, and one output neuron.

If the network has n input neurons, one hidden layer with m neurons, and one output neuron, the output of the network on the input vector $\bar{x}$ is given by

$$y = f\left(\sum_{j=1}^m w_j f_j\left(\sum_{i=1}^n w_{ji} x_i\right)\right), \quad (6.2)$$

where $f_j, j = 1, \dots, m$ and f denote the activation functions of the hidden layer neurons and the output neuron, respectively; $w_j, j = 1, \dots, m$ and $w_{ji}, j = 1, \dots, m, i = 1, \dots, n$ denote the weights of the output neuron and of the hidden neurons, respectively. The input vector $\vec{x} = (x_1, \dots, x_n)$ can also be extended by a unitary element for the bias of the weighting function.

The computing capabilities of multi-layer perceptrons, which are also referred to as feed-forward neural networks, are attributed by the two key features for complex static pattern recognition [140]:

Function approximation: feed-forward neural networks have been theoretically justified as universal approximators. They are, in theory, able to approximate any continuous function with arbitrary accuracy by the hierarchical architecture of simple processing units where the inputs are propagated to perform input-output mapping;

Training: feed-forward neural networks can be trained from the sample data using the error back-propagation algorithm, which is simple, accurate, and robust. If the network is trained properly, it is then able to respond correctly to previously unseen inputs with characteristic distributions similar to the training sample.

Feed-forward neural networks offer an immense scope for representing non-linear input-output mapping being specifically tuned by a number of adjusted weight values. The theoretical background for its mapping capability is based on the following theorem (for the case of n -dimensional space) [141]:

Theorem 1: There exist a set of constants w_j, b_j , and w_{ji} where $j = 1, \dots, m, i = 1, \dots, n$ and a bounded, smooth, and strictly monotonically increasing function f which maps the segment $[0;1]$ to itself, such that every continuous function $F(x_1, \dots, x_n)$ of n variables on $I_n = [0;1]^n$ can be approximated in the form:

$$y = \sum_{j=1}^m w_j f\left(\sum_{i=1}^n w_{ji} x_i + b_j\right), \quad (6.3)$$

where $|F(x_1, \dots, x_n) - y| < \varepsilon, \forall \varepsilon > 0$.

In the work by Cybenko [142], it was demonstrated that any multi-dimensional function can be approximated with any desired accuracy by the superposition of

sigmoid functions. The term “sigmoid” is used for the class of functions which satisfy the following constraints: $\lim_{x \rightarrow -\infty} f(x) = -1$, $\lim_{x \rightarrow +\infty} f(x) = 1$. Additionally, it was shown that a feed-forward neural network with a single hidden layer of $2n+1$ neurons, using sigmoid activation functions from the input to the hidden layer and from the hidden layer to the output layer, can compute any uniform approximation for a given set of inputs $(x_1, \dots, x_n) \in I_n$ and desired outputs $F(x_1, \dots, x_n)$.

Back-propagation learning.

Back-propagation is a supervised learning technique used to train artificial neural networks. It is mostly useful for feed-forward networks, which have no feedback. Back-propagation is used to calculate the gradient function of the network training error with respect to the network weights. The term "back-propagation" is also used in a more general sense, to refer to the entire procedure encompassing both the calculation of the gradient and its use in stochastic gradient descent. Back-propagation technique usually shows quick convergence on a satisfactory local minimum for training error in the kind of networks to which it is suited. The back-propagation learning strategy can only be applied to multilayer neural networks. The only requirement for the activation function is that it must be a differentiable sigmoid as in Theorem 1.

The back-propagation technique used for training of the FFNN designed for HRE recognition can be summarized as follows:

- i) Present an example HRE or a non-HRE sequence to the neural network with the corresponding desired output being ‘HRE’ or ‘non-HRE’ respectively represented in a numerical form.
- ii) Compare the network's output to the desired output for that example and estimate the local error. For each neuron, calculate the scaling factor, i.e. how its output must be adjusted to match the desired output.
- iii) Adjust the weights of each neuron to decrease the local error.
- iv) Assign "penalty" for the local error to the neurons at the previous level, giving higher “responsibility” to the neurons with higher weights. Repeat the steps above for the neurons at the previous level using each one's "penalty" as its error.

That is, the back-propagation technique for the FFNN training works as a “flow-back”. The weight adjustment for each output and hidden neuron is calculated as follows:

$$\vec{w}^{t+1} = \vec{w}^t + \alpha^t \times \delta \times \vec{x}, \quad (6.4)$$

where $\vec{w}^t$ is a vector of weights for the neuron at the t^{th} step of learning, α^t is the learning parameter at the t^{th} step ($0 < \alpha^t < 1, \forall t > 0$), and $\vec{x}$ is the input to the layer. The δ value for each neuron of the FFNN with one hidden layer and one output layer is the gradient descent function:

$$\delta^{output} = \left. \frac{\partial f(u^t)}{\partial u^t} \right|_{u^t = w^t x} \times (d^t - o^t) \quad \text{for the output layer, and} \quad (6.5)$$

$$\delta_h^{back-propagated} = \left. \frac{\partial f_h(u^t)}{\partial u^t} \right|_{u^t = w^t x} \times \sum_{k=1}^K w_{h \rightarrow k} \delta_k^{output}$$

for the hidden layer, (6.6)

where d^t and o^t are the desired and current outputs of the neuron respectively, $\vec{x}$ is the input to the layer (either hidden or output), $u^t = \vec{w}^{tT} \vec{x}$ is the synaptic input to the neuron, and $f(u^t)$ is the activation function of the neuron. For the back-propagated delta value, K is the number of neurons in the output layer, $w_{h \rightarrow k}$ is the weight coefficient of the connection between h^{th} neuron of hidden layer and k^{th} neuron of the output layer, and δ_k^{output} is the delta value for the k^{th} neuron of the output layer calculated as shown by the equation (6.5). The back-propagation learning process is terminated if either the error tolerance for the training accuracy is satisfied, or the maximum allowed number of learning cycles is exceeded, or the error plateau is reached meaning that no further improvement is available.

6.1.2 Network design and training

The trained FFNN for HRE recognition receives its input vector from the encoding module, and processes it through the sequence of layers. The output of the FFNN is a binary classification answer.

The neural network theory [139] suggests that for reliable results of learning, the number of the degrees of freedom, i.e. neuron weights to be fitted, has to be at most half the number of constrains (which are the training inputs accompanied with the desired outputs). Therefore, in the case of one hidden layer and the training set of approximately seven thousand positive and negative HREs, we should limit the number of hidden layer neurons to approximately 50. Thus, we will have $60 \times (50+1)$ weights of the hidden layer and $(50+1) \times 2$ weights of output layer (plus one is for the bias term), total 3'162. In the case of two hidden layers, the maximum number of neurons on each layer should be at most 40.

An important challenge of the FFNN training procedure designed for the two-class recognition problem comes from the following observation: the quantities of training vectors belonging to each of the classes must not be orders of magnitude different; otherwise, the network “forgets” about the smaller class and returns the correct answers for the larger class only. The second challenge to be considered is that if both training sets are large enough, in order for the FFNN to “keep in mind” both of them we have to mix them. Otherwise, while looking through several hundreds of functional HREs, the network adjusts itself for positive answers only. When the rest of the training set containing non-HRE pattern starts, the network “forgets” about the positive answers and trains itself for negative answers only, and so on. Such a switching of the learning strategy causes heavy oscillations during the training process and increases the chances to fall into an arbitrary local minimum.

For the binary classification required for HRE modeling, it is enough to represent the FFNN output as a single bit. However, in the proposed design, we represent the output of the neural network as a 2-vector for binary classification. In that case, the two-dimensional output space allows for higher flexibility for further decision making.

As confirmed by the experiments on supervised learning and testing, most of the input vectors come to convergence to an exact binary answer within floating point number precision, but the rest require setting up a threshold for decision making which is set as follows. The Euclidian distances between the actual network output and the exact outputs for binary classification are measured, and the output is converted to the

closest classifying vector if the distance to that is ten times smaller than that to the other. If the difference between the two distances is less than tenfold, then the output is marked as unknown.

For training the model, we use the back-propagation learning algorithm. Learning rate α in the equation (6.4) regulates the speed of convergence of the gradient descent: the higher is the learning rate, the faster system converges. However, if the learning rate is excessively high, then there is a high chance to jump over the minimum of error, or fall into oscillations. Our solution is to adjust the learning parameter dynamically: if the training error at the current learning cycle is less than that at the previous cycle, then the learning process is heading in a right direction, so we increase the learning rate by 5%. If the learning process jumps over a minimum and exhibits larger errors, then α is decreased by 30% resulting in further smaller changes of neuron weights. The learning procedure is stopped if either the 99.9% accuracy level is exceeded, or the maximum number of back-propagation cycles (which is at most 5000), or the error plateau is reached, meaning that no improvement of the training error has been detected over one hundred of consecutive operations.

6.1.3 Prediction results

The set of collected functional HREs was used for training and testing. Similar to previous models, this dataset was split into training and testing parts in different proportions, and overlapping vs. non-overlapping separations were tested.

Table 6.1 summarizes the FFNN operation results, namely, the training parameters and the resulting accuracy of HRE prediction. In this table, the four possible training-testing proportions are given, and for each of them, the corresponding accuracy and computational complexity values are shown.

Different neural network layouts were examined. In particular, the neural networks with one hidden layer containing 20, 40, and 60 hidden neurons were compared to the network with two hidden layers. In the second column of table, the letter I denotes the network input layer, the letter O denotes the output layer, and the size of one or two hidden layers is shown in between I and O. Additionally, the median number of required learning cycles was tracked during the series of randomly generated training-testing trials for each training-testing proportion value, and the median computational time to the error plateau was estimated as well. The median values are selected instead of means because the plateau theoretically may be not

reached if the network is poorly designed or falls into oscillations while learning.

Table 6.1 Results of HRE prediction by the FFNN.

Training/ testing	Network layout	Learning error, %	Testing error, %	RE, bp	Learning cycles	Training time, sec
50/50	I $\rightarrow$ 20 $\rightarrow$ O	3.0	2.5	4790	188	120
70/70		3.1	2.0	4800	195	160
70/30		3.1	2.6	4800	195	160
100/100		2.6	2.6	4950	258	240
50/50	I $\rightarrow$ 40 $\rightarrow$ O	2.0	2.5	5350	89	~200
70/70		2.1	2.0	5300	95	~300
70/30		3.1	2.8	5300	95	~300
100/100		2.6	2.6	5150	96	~400
50/50	I $\rightarrow$ 60 $\rightarrow$ O	2.0	2.5	5550	39	~240
70/70		2.0	2.0	5450	49	~340
70/30		2.0	2.9	5450	49	~340
100/100		2.6	2.6	5300	50	~470
50/50	I $\rightarrow$ 30 $\rightarrow$ 20 $\rightarrow$ O	2.0	2.5	5650	75	~300
70/70		3.1	2.0	5650	77	~450
70/30		3.1	2.6	5650	77	~450
100/100		2.6	2.8	5600	81	~600

Fig. 6.2 The ROC curve for HRE prediction by the FFNN with layout I $\rightarrow$ 40 $\rightarrow$ O.

For further development, the network layout with one hidden layer containing 40 neurons is used. The AUC for such FFNN is 0.981.

6.2 A Hopfield Neural Network for Structure-Based HRE Classification

Hopfield neural network (HNN) is an intelligent computational method suitable for classification in the case of partially overlapping classes. In particular, HNN may be used for recognition of motif structures which are sometimes cross-clustered. The applicability of recurrent neural networks for biological sequence analysis has recently been reviewed by Hawkins and Boden [33] with an example of motif identification and prediction of sub-cellular localization of peptides. The empirical results confirm that though the network architecture reflects the presence of bias, the recurrent neural networks indeed provides access to the patterns of biological significance. Inspired by these findings, we adapt a recurrent neural architecture for the problem of symmetric structure prediction, and, in particular, for prediction of the HRE dimeric structure.

In this section, we present a novel idea for improved recognition of symmetrically structured DNA motifs by enhancing the specificity of the previously designed highly sensitive methods. The developed HNN is aimed to consider the dimeric structure of putative HREs and thus eliminate false positive predictions. The idea is motivated by numerous experimental reports which demonstrate that the nature of interaction of hormone receptors with DNA in promoter regions of target genes is the dimerization. The HNN capable of accurate modeling the HRE dimeric structure should increase the selectivity of any prediction algorithm.

6.2.1 Symmetric structure of HRE

The three-dimensional structure of the DNA binding domain (DBD) of hormone receptors was established by nuclear resonance spectroscopy [143, 144]. The crystal structure of the steroid receptor DBD in complex with DNA was determined as well [58, 71]. Based on these findings we know that the important feature of the DBD is its Zn coordination sites. Namely, the two Zn ions are tetrahedrally coordinated by four cysteines to stabilize two peptide loops and cap amino termini of two amphipathic α -helices. Residues that coordinate the Zn atoms and support the fold of the domain are conserved throughout the family, and this fact confirms that the Zn module structures are also conserved among the receptors [53].

The loops of both Zn modules form phosphate backbone contacts. The DNA recognition helix resides within the loop region, thus forming the specific nature of

protein-DNA interaction via the so called “zinc fingers“. The structural similarity of the two Zn finger modules gives the receptor protein the appearance of an approximate structural repeat. Moreover, the zinc finger modules are encoded by separate exons. This fact suggests that the modules may have arisen due to DNA duplication.

The symmetric nature of HREs known since long confirms that hormone receptors bind its DNA targets as dimers. Furthermore, the center-to-center separation of half-sites in the HRE motif has nine nucleotide base pairs which is nearly a helical repeat, suggesting that the two subunits are located on nearly the same face and can contact each other. Indeed, crystallographic analysis [71] and EMSA studies [64, 145] show it to be true for the glucocorticoid receptor DBD. Moreover, in addition to the dimeric nature of the protein-DNA interaction, the dimerization may have different structure, namely head-to-head, head-to-tail, or tail-to-tail composition [146], thus providing us the reason to consider different orientations of the half-sites for modeling HRE dimeric structure.

The schematic representation of the dimeric protein-DNA interaction is given in Fig. 6.3, and the HRE sequence profile obtained using our collection of experimentally validated HREs is given in Fig. 6.4.

Fig. 6.3 The HRE dimeric structure.

In Fig. 6.3, the HRE dimeric structure is shown as a direct or an inverted repeat, and additionally, instead of considering the complementary DNA strand for HRE prediction, we consider the two possible orientations of the HRE structures on the same DNA strand. In Fig. 6.4, we may notice the high conservation of the expected contact points for the receptor-HRE interaction (refer to Section 2.2 for description). Another observation is the low selectivity towards the orientation of the half-sites, especially the left one, which is known to be less conserved (for the review of current knowledge about HRE sequence composition see Section 3.1). Note that the contact points remain the same for both direct and reverse orientations of the half-sites.

Fig. 6.4 Sequence logo representation of the steroid HRE.

Hence, we incorporate the idea of dimeric structure into the HRE recognition model and assume that if a predicted HRE sequence cannot be assigned with a symmetric repeated composition, then it is unlikely involved into the dimeric protein-DNA interaction.

6.2.2 Recurrent neural model

Recurrent neural networks were initially developed for sequence recognition tasks in the domain of natural language processing [147]. They were also shown to serve effectively as associative memories [140] and pattern classifiers [148–150].

A recurrent neural network is a neural network where the connections between the units form directed cycles. Recurrent neural networks must be treated differently from feed-forward neural networks, both when analyzing their behavior and training them. Usually, theory of dynamical systems is used to model and analyze the recurrent neural networks. A popular type of recurrent neural networks is the Hopfield network, though there exist other types of recurrent neural networks, namely the two-layer Elman network [151], which is a combination of a recurrent and a feed-forward data

processing schemes.

The behavior of recurrent neural networks is more like that of an iterative process or a dynamical system rather than a neural network in its conventional feed-forward form. The equations that describe the recurrent network operations are defined as follows:

$$\vec{a}(0) = \vec{\xi} \quad (6.7)$$

$$\vec{a}(t+1) = F(W \cdot \vec{a}(t) + \vec{b}) \quad (6.8)$$

where $\vec{\xi}$ is the input vector to the neural network, F is the activation function, $\vec{a}(t)$ is the output vector of the network after t^{th} cycle, W is the matrix of neuron weights for the recurrent layer, and $\vec{b}$ is the vector of biases of the network.

In order to apply the theory of dynamical system stability to the recurrent neural networks, some additional assumptions about the matrix W must be made:

$$w_{i,i} = 0, \forall i = 1..N, \quad (6.9)$$

$$w_{i,j} = w_{j,i}, \forall i, j = 1..N. \quad (6.10)$$

That is, the connections are symmetric, and no recurrent neuron has a connection to itself. Here, N is the size of the recurrent layer. A recurrent neural network with such limitations is conventionally referred to as the Hopfield neural network (HNN) [152]. The requirement that weights are symmetric is to guarantee that the error function decreases monotonically while following the activation rules, and the system reaches a stable state after a finite number of learning cycles:

$$\frac{d\vec{a}(t)}{dt} = 0, \quad t > T, T < +\infty. \quad (6.11)$$

For the problem of HRE recognition, the stable states for HNN design are the HRE dimeric structures, and the HNN is expected to reach a stable state which corresponds to one of those structures with any input DNA sequence used as an initial

condition. If the network comes to a stable state other than the set of HRE structures or causes infinite oscillations, then the input DNA sequence is defined as misclassified. Otherwise, a certain dimeric structure is attributed to the input DNA sequence. The theory and algorithm of HRE structure prediction by the HNN are described in details in the next sub-section.

Figure 6.5 shows an example of HNN operation on a 2D $[-1;1]^2$ hypercube. The Hopfield network has A) two stable states; B) three stable states; C) three pre-defined and one spurious stable state. The crosses are the network initial conditions, the circles represent the stable states, and the lines are the trajectories of the network convergence.

Fig 6.5 Convergence of the HNN in a 2D space.

Dynamic system-based training

The goal for the training of the recurrent neural network is to design a discrete system that possesses a specific set of equilibrium points such that, when an initial condition is provided, the network eventually converges at such an equilibrium point. The network is recursive in that the output is fed back as the input once the network is in operation. Hopefully, the network output eventually settles in one of the original design points.

When the prototype stable state patterns are orthogonal, the problem in describing a corresponding HNN can easily be solved by the Hebb rule [139]. Otherwise, a more complex procedure of neural system synthesis must be used. Accurate numerical approximation of an iterative discrete machine by a dynamical system has been proposed by Li et al. [153]. We adapt the proposed approach to learning of the HNN that is designed for recognition of symmetrically structured HRE

motifs.

Suppose we are given m vectors that represent the asymptotically stable equilibrium points for an N -dimensional dynamic system: $\vec{\xi}_1, \dots, \vec{\xi}_m \in \mathfrak{R}^N$. We proceed the HNN learning as follows:

1) Compute $N \times (m-1)$ matrix Z :

$$Z = [\vec{\xi}_1 - \vec{\xi}_m, \dots, \vec{\xi}_{m-1} - \vec{\xi}_m] \quad (6.12)$$

2) Perform a singular value decomposition of Z and obtain the matrices U , V , and S such that $Z = USV^T$, where U and V are unitary matrices, and S is a diagonal matrix.

$$U = [\vec{u}_1, \dots, \vec{u}_L]. \quad (6.13)$$

Let k be the rank of Z (and U correspondingly).

3) Compute the auxiliary matrices

$$T^+ = \sum_{i=1}^k u_i u_i^T, \quad (6.14)$$

$$T^- = \sum_{i=k+1}^L u_i u_i^T, \quad (6.15)$$

$$T_\tau = T^+ - \tau \cdot T^-, \quad (6.16)$$

$$E_\tau = \vec{\xi}_m - T_\tau \cdot \vec{\xi}_m, \quad (6.17)$$

where τ is a learning parameter. According to the proposed approximation theory, the larger is the value of τ , the less spurious stable states the reconstructed HNN has. For synthesis of the HNN with more than three stable states, a value $\tau \geq 10$ is usually used.

4) Compute the parameters of the approximating dynamic system $\vec{a}((t+1)h) = F(W \cdot \vec{a}(t \cdot h) + \vec{b})$, namely W and $\vec{b}$, as follows:

$$W = U \begin{bmatrix} e^h I_k & 0 \\ 0 & e^{-\tau h} I_{L-k} \end{bmatrix} U^T, \quad (6.18)$$

$$\vec{b} = U \begin{bmatrix} (e^h - 1) I_k & 0 \\ 0 & \frac{1}{\tau} (e^{-\tau h} - 1) I_{N-k} \end{bmatrix} U^T E_\tau, \quad (6.19)$$

where I_k is the $k \times k$ identity matrix. The parameter $0 < h < 1$ reflects the asymptotic nature of the procedure, since discrete states of the neural network are replaced by continuous values of the dynamic system.

The trained HNN is then used for HRE structure prediction. For each putative HRE used as an input, it returns the output which represents a dimeric or a non-dimeric HRE structure, or a misclassification label.

6.2.3 Hopfield neural network for modeling the HRE dimer

Mathematically, the problem of recognizing highly diverse TFBSs in DNA can be tackled using the notion of phase spaces. We consider this problem by the example of HREs, though it can be extended to the cases of other symmetrically structured TFBSs.

Let the space of functional HREs be a subspace of all DNA sequences. This subspace should be distinguishable from other sequences, at least because the hormone receptors are capable of distinguishing their binding sites from neutral DNA. Thus, we find a set of basis vectors (which should serve as major indicators of a functional HRE) in this subspace, and then use a Hopfield neural network with the same basis vectors as stable states, for HRE structure prediction. Convergence of the network from its initial condition to a stable state corresponding to a functional HRE structure serves as an indicator whether the input vector belongs to the subspace of functional HREs or not. A functional HRE which is different from the consensus HRE can be considered as a disturbance in the space of HREs (whose diversity is usually explained by evolution and neutral mutations), and thus should be smoothly converted into one of the basis vectors by the trained HNN. Neutral DNAs are considered as the elements of the complimentary subspace.

Molecular biology provides us with prior knowledge of various possible appearances of HREs *in vivo* (refer to Sub-section 6.3.1). We use this information to

construct the subspace of interest. In the case of dimeric HREs, the various characteristics of a functional response element are the orientations of HRE half-sites and the presence of a non-conserved flexible spacer.

Consider the consensus HRE dimer with the half-site $\vec{y}$ of b base pairs length $\vec{y} = [y_1, \dots, y_b]$, $y_j \in \Omega$, $j = 1, \dots, b$, and the spacer δ_c of c nucleotides. The possible HRE dimeric structures can be represented as follows:

$$\begin{aligned} s_{DR} &= \vec{y} \cup \delta_c \cup \vec{y} \\ s_{IR} &= \vec{y} \cup \delta_c \cup \bar{\vec{y}} \\ s_{PR} &= \vec{y} \cup \delta_c \cup \vec{y} \\ s_{ER} &= \vec{y} \cup \delta_c \cup \bar{\vec{y}} \end{aligned} \quad (6.20)$$

where $\bar{\vec{y}} = [y_b, \dots, y_1]$ represents the reverse of $\vec{y}$. Also, s_{DR} is called a direct repeat, s_{IR} is an inverted repeat, s_{PR} is a palindromic repeat, and s_{ER} is an everted repeat. The subspace of dimeric HRE sequences is therefore defined as $\Psi = \{s_{DR}, s_{IR}, s_{PR}, s_{ER}\}$, and the Hopfield neural network $\aleph$ performs the transformation from the space of HRE sequences Θ into the space of HRE structures Ψ :

$$\aleph : \Theta \rightarrow \Psi. \quad (6.21)$$

Due to the necessity of numerical representation for the input data, the Hopfield neural model is forestalled by the same one-hot encoding module $\Sigma : \Theta \rightarrow \Xi_\Theta$ as described in Section 6.1 for the feed-forward neural network. Hence, the HNN designed for dimeric structure prediction works as follows:

$$\aleph : \Theta \xrightarrow{\Sigma} \Xi_\Theta \xrightarrow{HNN} \Xi_\Psi \xrightarrow{\hat{\Sigma}} \Psi \quad (6.22)$$

where HNN stands for the Hopfield Neural Network operation unit.

Based on the current knowledge about the HRE composition, we consider the consensus DNA sequence of the HRE half-site to be $\vec{h}_{HRE} = \text{TGTTCT}$ [55, 61, 154]. With reference to the above notation, the four possible structures of HRE are

$$\begin{aligned}
HRE_{DR} &= [\bar{h}_{HRE} \cup \omega_{sp1} \omega_{sp2} \omega_{sp3} \cup \bar{h}_{HRE}] \\
HRE_{IR} &= [\bar{h}_{HRE} \cup \omega_{sp1} \omega_{sp2} \omega_{sp3} \cup \bar{h}_{HRE}] \\
HRE_{PR} &= [\bar{h}_{HRE} \cup \omega_{sp1} \omega_{sp2} \omega_{sp3} \cup \bar{h}_{HRE}] \\
HRE_{ER} &= [\bar{h}_{HRE} \cup \omega_{sp1} \omega_{sp2} \omega_{sp3} \cup \bar{h}_{HRE}]
\end{aligned} \tag{6.23}$$

where $[\omega_{sp1} \omega_{sp2} \omega_{sp3}]$ stands for the 3bp-long spacer ($\omega_{sp,i} \in \Omega$, $\forall i=1,2,3$). These consensus HRE structures and six non-HRE sequences (taken from the experimental papers by Thackray et al. [155] and by Lieberman et al. [60]) are used as ten equilibrium points for the HNN design.

6.3 Global Optimum of HRE Structure Prediction with Dynamic Programming

In the previous section, we described the method for predicting symmetric structure of DNA motifs by a trained Hopfield neural network. After a series of preliminary tests, we found that the HNN model works well for the case of dimeric HRE motifs [156] and allows to eliminate a large amount of false positive HRE prediction. However, the exact-match nature of the approach, which implies the rigid correspondence between the input and output vector elements, adds serious artifacts into the HRE modeling process. That is, in the exact-match HNN, we look for the recurrent convergence between the putative HRE sequence and its dimeric structure considering the input and the output vectors explicitly matched against each other, similarly to the exact match sequence comparison. However, real biological sequences are exposed to short mutations including nucleotide insertions and deletions (indels) which are not always critical for further interaction with proteins [60]. For higher flexibility of HRE modeling, we have to incorporate short indels into the dimeric structure prediction.

The incitement for the model enhancement comes from the following observation: while single nucleotide mismatches are successfully handled by the HNN recurrence process, short indels can bring significant distortion into the model. In particular, as a result of an insertion of a random nucleotide into the HRE spacer (with no other changes in the rest of HRE sequence), the predicted dimeric structure varies for 14% of functional response elements. Additionally, when a single nucleotide is

deleted from the spacer (also with no other changes), the predicted dimeric structures varies for 9% of the training data. Finally, the dimeric structure predicted by the trained HNN is affected by a single nucleotide indel within the HRE spacer for 18% of functional HREs (119 out of 661 response elements).

These observations motivate us to consider the reliability of the recurrent system for structure prediction more carefully, because from biological point of view such oscillating predictions are meaningless. In particular, the lattice of dimeric protein-DNA interaction is expected not to be influenced by 1bp difference in the spacer length, even though the HRE binding affinity may indeed change [64]. Thus, the necessity of developing a more flexible model for dimeric structure prediction is confirmed by the observed lack of robustness when modeling the complex HRE pattern. On the other hand, single nucleotide substitutions within the half-sites or spacers (1500 random nucleotide substitutions in 661 HRE sequences) do not cause predicted dimeric structures to vary for 99% of functional HREs.

For the classic exact-match problem of sequence comparison which has a similar disadvantage, numerous sequence alignment methods are proposed. These methods successfully deal with short insertions and deletions which appeared as a result of point mutations while not affecting the similarity and functionality of biological sequences, and therefore make sequence comparison more robust.

6.3.1 Dynamic programming for alignment of biological sequences

In bioinformatics, sequence alignment is a way of arranging the DNA, RNA, or protein sequences to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. At early stages of biological science, the sequences were checked for exactly matching regions, but as more information and functional relationships were discovered, the limitations of the exact match approach for sequence analysis became crucial.

Indeed, biological sequences are usually subjected to short mutations resulting in inter- and intra-species diversity. In the case of exact match comparison, every single nucleotide insertion or deletion shifts the frame for sequence comparison and leads to meaningless results. Due to this reason, the alignment techniques are used for biological sequence comparison instead of the exact-match screening.

The alignment procedure attempts to represent the evolutionary relationship between two DNA, RNA, or protein sequences by placing them side-by-side, so that

the nucleotides or amino acids paired in an alignment have arisen from a single nucleotide or amino acid in a single ancestral gene or protein with the highest possible probability.

The first efficient global alignment algorithm based on dynamic programming was proposed by Needleman and Wunsch in early 70s [157]. This algorithm, named after its authors, provides a globally optimal solution for the alignment of two or more sequences in terms of the resulting alignment score, which benefits from as many matches as possible. Though gaps within the aligned sequences usually decrease the total alignment score, the resulting match rate may increase significantly if a short frame-shifting mutation, namely, an insertion or a deletion, indeed has taken place during the historical process.

The dynamic programming alignment requires constructing the n -dimensional array, or so called “hyper-matrix” (i.e. matrix in the n -dimensional space), where n is the number of sequences to be aligned. Although this technique is computationally expensive, it guarantees finding a global optimum solution for any given scoring scheme. For the case of long sequences, certain heuristic optimizations are used in order to decrease space and time complexity. However, dealing with relatively short sequences like protein binding motifs on DNA we usually can afford exponential space and computational time. That is, this approach is particularly useful when a few short sequences need to be aligned accurately.

We adapt the dynamic programming approach used for Needleman-Wunsch alignment, for a more flexible matching between the HRE sequence and its prototype dimeric structure. The enhanced matching strategy is expected to increase both the robustness of structure modeling and the sensitivity of HRE prediction.

6.3.2 The DP-HNN model

During the HNN operation, single nucleotide substitutions within HRE sequences are successfully absorbed by the convergence process. Per contra, every insertion or deletion inside the HRE sequence causes a frame shift of a half of the sequence on average, so that the convergence for the rest of the sequence becomes misleading. We overcome this exact-match limitation involving the method of dynamic programming similar to that used for sequence alignment.

For the problem of HRE structure prediction by the HNN, the exact matching between input and output is per se implied by the weight matrix by the input vector

multiplication procedure (refer to the equation (6.8)).

Extending this multiplication, we have the following identity:

$$W \cdot \vec{a}(t) = \left\| \sum_{j=1}^{4L} w_{i,j} a_j(t) \right\|_{i=1..N}, \quad (6.24)$$

where the assumption of j^{th} element of the input vector $\vec{a}(t)$ being converted into j^{th} element of either S_{DR} , or S_{IR} , or S_{PR} , or S_{ER} dimeric structure is implied by the multiplication $w_{i,j} a_j(t)$ of the j^{th} weight by exactly the j^{th} element of the input vector for each i^{th} neuron of the HNN. Thus, each nucleotide insertion or deletion can be treated as a submission of the j^{th} element of the input vector to the $(j \pm shift)^{th}$ neuron, and therefore can be represented in the form of index shifts for weight vectors for the procedure of multiplication. The entire procedure of establishing the best matching between the current input sequence and its dimeric prototype is referred to as the *recurrent alignment*.

For the problem of recurrent alignment of HREs, the input vector $\vec{a} = \vec{a}_1 \cup \omega \cup \vec{a}_2$ is composed of the two half-sites $\vec{a}_1$ and $\vec{a}_2$ of length $4(b+1)$, and a single nucleotide ω located in the center. The elements of the half-site vectors are numbered starting from the internal nucleotides, and unlike the previously described HRE representation, we merge two nucleotides of the spacer each to the nearest half-site.

The alignments with the two possible orientations of the consensus HRE half-site $\vec{y}$ and $\vec{\bar{y}}$ are performed using pre-defined gap penalties, and the best alignment is then selected as a prototype for the input half-site convergence. For accomplishing the alignment procedure, for each $\vec{a}_2$ and $\vec{a}_1$, two scoring matrices are generated:

$$M_{\vec{y}}(k_1, k_2) = \begin{cases} M_{\vec{y}}(k_1 - 1, k_2 - 1) + s(a_{k_1}, y_{k_2}) \\ M_{\vec{y}}(k_1 - 1, k_2) + g_a \\ M_{\vec{y}}(k_1, k_2 - 1) + g_h \end{cases} \quad (6.25)$$

(and similar $M_{\bar{y}}$ for the reverse consensus half-site $\bar{y}$), where

$$s(a_{k_1}, y_{k_2}) = \begin{cases} s > 0, & \arg \max_{i=1..4} \xi_{4(k_1-1)+i}^{\bar{a}} = \arg \max_{i=1..4} \xi_{4(k_2-1)+i}^{\bar{y}} \\ 0, & otherwise \end{cases} \quad (6.26)$$

and $g_a < 0$, $g_h < 0$ are the gap penalties for deletion and insertion in the input vector respectively.

Vector $\vec{\xi}^{\bar{y}}$ is calculated as follows:

$$\vec{\xi}^{\bar{y}} = \Sigma(\bar{y}), \quad (6.27)$$

where Σ is the DNA encoding module (refer to Section 6.2). In the equation (6.26), equivalence of $\arg \max_i$ functions denotes that the two 4-vectors for the considered nucleotide positions have maximums at the same internal positions. If the two $\arg \max_i$ functions are equal, the nucleotide element of the input sequence is considered as a match with the considered nucleotide of the dimeric structure.

Initial conditions are selected so that the spacer length of 3 base pairs is preferred to any other length, as it has been shown in numerous experiments for steroid hormones [130]. Thus,

$$\begin{aligned} M_{\bar{y}}(0,0) &= 0 \\ M_{\bar{y}}(k,0) &= sp + g_s \cdot (k-1) \\ M_{\bar{y}}(0,k) &= g_s \cdot k \end{aligned} \quad (6.28)$$

(and the same for the matrix $M_{\bar{y}}$) where $sp > 0$ represents a positive score for one nucleotide insertion at the beginning of the alignment for each half-site. This insertion, when joined with the central nucleotide ω and a similar insertion from the other half-site alignment, results in exactly 3bp-long spacer. Additionally, $g_s < 0$ is the gap penalty for a longer spacer.

The resulting alignments for the input half-sites are obtained by tracking back the

scoring matrices (one of two calculated for each half-site) which has the largest values of their last columns and last rows. The resulting alignment of the input vector constructed from the alignments of its half-sites is then denoted as $\vec{al}(t)$. Its length N_{al} holds the inequality $L_{al} \leq 2L$ where L is the length of HRE pattern.

That is, instead of exact correspondence "the j^{th} element of input $\leftrightarrow$ the j^{th} element of output", we obtain a dynamically adaptable correspondence between the input sequence and its target output structure. The current output of the HNN is thus calculated as follows:

$$W \otimes \vec{a}(t) = \left\| \sum_{j=1}^{2b+1} w_{i,j+4deletion(j)} a_{j+4insertion(j)}(t) \right\|_{i=1..N} \quad (6.29)$$

Here, the operation $\otimes$ denotes the "aligned" multiplication, i.e. multiplication involving corresponding weight shifts.

Shifts indices $deletion(j)$ and $insertion(j)$ for the neuron weights are calculated from the alignment $\vec{al}(t)$ of the input vector $\vec{a}(t)$ using the following procedure:

```

deletion(0) := 0;
insertion(0) := 0;
for i = 1:L
    deletion(i) = deletion(i-1);
    insertion(i) = insertion(i-1);
    if al(i + deletion(i-1)) == 'deletion'
        then deletion(i)++;
    else
        if al(i + deletion(i-1)) == 'insertion'
            then insertion(i)++;
    i++;

```

In equation (6.29), we multiply each index shift by 4 since in one-hot notation, each nucleotide corresponds to exactly four elements of the network input vector.

Figure 6.6 illustrates the procedure of recurrent alignment by the example of the right half-site of the input sequence 'AAAAAAAAGTGATTT' being aligned with the consensus HRE half-site 'TGTTCT'. In this example, the insertion and deletion penalties are $g_a = g_h = -2$, the long spacer penalty is $g_s = -1$, the consensus spacer and the matching score are $s = sp = 2$. The alignment with one deletion and one insertion, in addition to one insertion into the expected spacer, is shown. For clarity, the nucleotide bases instead of the 4-vectors are used. With weight shifts, after the first insertion, the 2nd element of the aligned right half-site, namely, T, (the 5th-8th elements of the transformed vector) is an input to the 1st quartet of neurons for the given half-site (neurons 1-4). After the second insertion, the 4th element of the aligned right half-site, namely, T, (13th-16th elements) is an input to the 2nd quartet of neurons, and so on.

Fig. 6.6 An example of recurrent alignment. For clarity of the picture, the dotted arrows represent sets of omitted HNN connections.

Thus, each HNN operation cycle is accompanied with the recurrent alignment procedure of the current input sequence and its targeted structure. This architecture provides us with a more flexible dimeric structure modeling scheme and is expected to result in more robust, hence, more reliable predictions.

6.3.3 Dimeric structure prediction for the collection of functional HREs

We experimentally evaluated the performance of the proposed HNNs on assigning the functional response elements from our dataset with HRE dimeric structures. Another aim was to find any interesting trends in structural preferences for the collected data set of HREs, and also test the applicability of the approach to the general problem of modeling the symmetrically structured weak TFBS signal.

We tested both the dynamically adaptable (DP-HNN) and the exact match (EM-HNN) versions of the Hopfield neural network. Additionally, since the set of possible dimeric and non-dimeric structures is pre-defined, the HNN models can be considered as sequence classifiers; thus, we performed k-means classification of the same set of HREs for comparison. Unlike the HNN classifiers with fixed stable states, the k-means procedure iteratively adjusts the set of its centroids, so that the cumulative variance for these centroids and the points in the dataset is eventually minimized. That is, for the procedure of k-means classification, instead of fixing the class centroids we set the possible dimeric structures as starting points for centroid adjustment.

The results of the three classification procedures for the set of functional HRE sequences are shown in Table 6.2. In this table, the first column is the total number of HREs for a given steroid hormone receptor (namely, progesterone, glucocorticoid, or androgen receptor). In the second column, we list the possible HRE dimeric structures for each of the groups. The next columns show the results of dimeric structure prediction by a given classifier for the HRE group.

As Table 6.2 shows, the three hormone receptors of interest demonstrate different preferences towards dimeric structure of their response elements on DNA. The difference between the distributions of predicted HRE structures for ARE, PRE and GRE is statistically significant ($p\text{-value} < 10^{-4}$) for k-means clustering, and for Hopfield-based classification as well ($p\text{-value} = 0.007$). However, this finding is not unexpected since a similar observation that AREs has stronger preferences towards the IR structure has already been reported by Reid et al. [158] and Claessens et al. [159].

Table 6.2 Results of dimeric structure prediction by the HNN and k-means.

HRE	HRE Structure	EM-HNN		DP-HNN		k-Means	
		N	%	N	%	N	%
PRE total: 66	DR	35	53.0	32	48.5	26	39.4
	IR	1	1.5	2	3.0	5	7.6
	PR	24	36.4	27	40.9	20	30.3
	ER	3	4.5	4	6.1	12	18.2
	non-HRE	3	4.5	1	1.5	3	4.5
GRE total: 377	DR	225	59.7	210	55.7	134	35.5
	IR	3	0.8	7	1.9	26	6.9
	PR	90	23.9	124	32.9	127	33.7
	ER	28	7.4	22	5.8	62	16.4
	non-HRE	31	8.2	14	3.7	28	7.4
ARE total: 218	DR	94	43.1	93	42.7	33	15.1
	IR	1	0.5	5	2.3	55	25.2
	PR	65	29.8	67	30.7	69	31.7
	ER	52	23.9	51	23.4	40	18.3
	non-HRE	6	2.8	2	0.9	21	9.6
U total 661	DR	354	53.6	335	50.7	193	29.2
	IR	5	0.8	14	2.1	86	13.0
	PR	179	27.1	218	33.0	216	32.7
	ER	83	12.6	77	11.6	114	17.2
	non-HRE	40	6.1	17	2.6	52	7.9

Notation: N – number of HREs, % – portion of HREs for a hormone receptor.

In order to estimate the robustness of the dynamically adaptable neural model, we performed a series of tests where single nucleotides were inserted to or deleted from the spacers and half-sites of the HRE sequences. The assumption under test was that, as a result of an 1bp indel, the prediction outcome might only vary from the HRE to non-HRE structure or visa versa (reflecting a possible change of binding affinity). At the same time, there should not have been an exchange of prediction from one HRE structure to another, because such an alteration is unlikely to occur when zinc-fingers of the receptor's DNA-binding domain interact with dimeric DNA binding sites *in vivo*.

The results of the tests revealed that using the dynamic adaptation indeed increases the robustness (and reliability) of our Hopfield neural system. We observed

that 1 indel within the HRE spacer caused variation of structure prediction returned by the EM-HNN for 18% of HREs, while for the DP-HNN this variation was only 3%. Additionally, one indel within half-sites is critical for 7% of predictions made by the EM-HNN, and for 1% of predictions made by the DP-HNN.

However, in spite of these findings, there still exists a small group of HREs, which are highly different from other known HREs and cannot be robustly assigned with a dimeric structure, though have been convincingly proved to be functional (reviewed in [160]). If those outlying HREs are not false positives, then the nature of their interaction with the hormone receptor's DBD should be considered more carefully.

6.4 A Two-Phase Neural Model for HRE Recognition

In the previous sections of this chapter, we described the two different ways to model HREs, namely, the HRE sequence modeling using the feed-forward neural network, and the HRE structure modeling using the Hopfield neural network. However, we expect the ensemble of diverse predictors to be more accurate than standalone methods. Therefore, we design an enhanced HRE recognition procedure that involves a cascade of these two neural networks. The feed-forward neural network described in Section 6.2 is used for recognition of HRE-like sequences. The Hopfield neural network described in Sections 6.3-6.4 is then used to classify the predicted HREs into one of five dimeric structure groups, namely, the two direct repeats, two inverted repeats, or a non-HRE. If a sequence is marked as an HRE-like, but cannot be reliably assigned with any dimeric structure, it is unlikely involved in homodimeric DNA-protein binding.

We used the collected dataset of functional HREs for training and testing purposes. For the negative control, the set of ten 1Mb-long neutral DNA sequences was generated with the assumption of independent equally distributed nucleotide content. Though a TFBS random expectation rate may theoretically be influenced by the involvement of dinucleotide composition as of real genomes for some transcription factors, the distribution of HRE motifs is known not to be affected by the assumption of the context nucleotide independence [136].

In our implementation, the procedure of recognizing the HRE motifs in DNA operates as shown in Fig. 6.7:

1. Design of the two-phase neural system

a) The feed-forward neural network with one the hidden layer of 40 neurons and the two-neuron output layer is trained using the set of experimentally validated HREs and the tenfold set of neutral DNA sequences. The output of this network is a 2-vector with probabilities of being an HRE and a non-HRE for each input. The balance of these two values for the following decision making system is subject to a threshold;

b) Given the set of HRE dimeric structures $\Psi = \{s_{DR}, s_{IR}, s_{PR}, s_{ER}\}$ transformed by the encoding module $\Xi_{\Psi} = \Sigma(\Psi)$, we construct the HNN system for HRE structure prediction. Only positive outputs of the previous sequence-based HRE recognition by the FFNN are processed by the HNN (unless stated otherwise, e.g. for speed testing purposes). The output of that system for each putative HRE input is either one of four possible dimeric structures, or the “non-dimer” output.

2. Recognition of HREs

To recognize HREs in DNA sequences, we use a sliding window of a fixed length of 15bp to obtain the stream of DNA subsequences which are then processed by the two-phase system.

Fig. 6.7 A two-phase neural model for HRE Recognition.

The proposed HRE recognition process operates as follows. First, a transformed DNA subsequence $\tilde{\xi} = \Sigma(\tilde{\theta})$ is submitted to the FFNN, which returns the probability of this subsequence to be an HRE to the decision making module. Second, the list of

putative HREs returned as a result of operation of the FFNN is processed by the HNN, and for each subsequence its dimeric structure is predicted. The output of the system is either a binary answer for each input sequence, or a list of predicted HREs with their annotations if the screening of a long DNA region is performed.

Table 6.3 shows the results of HRE prediction by the described two-phase neural system in comparison with the standalone FFNN. The layout of the neural architecture is shown in the first column with arrow denoting the succession of neural networks. Human RE (random expectation) is the prediction rate for the human genome (NCBI Genbank #36.2).

The first step of HRE prediction machine learning is awakening the trained feed-forward neural network which predicts HRE-like patterns. As demonstrated, for the FFNN step the prediction sensitivity value is as high as 98% (i.e. 15 among 661 HREs were always misclassified), combined with the specificity of 1:6Kb.

Table 6.3 Accuracy of HRE prediction by the two-phase neural model.

Neural network	Misclassified HREs	Sensitivity, %	Specificity, kbp ⁻¹	Human RE, kbp ⁻¹	AUC
FFNN	15	98±4.4%	5.84±0.78	7.28	0.98±0.04
FFNN→EM-HNN	52	92±2.3%	7.29±1.13	8.15	0.92±0.03
FFNN→DP-HNN	25	96±2.6%	7.08±1.21	8.14	0.96±0.03

The Hopfield neural network allows increasing the specificity level to 1:7.3Kb, while the sensitivity is kept at the reliably high level of 92% (6 or 9% of PREs, 37 or 10% of GREs, and 9 or 4% AREs, or total 52 HREs, are misclassified).

Two interesting observations from the Table 6.3 can be made. First, the area-under-curve for HRE prediction by solely FFNN is better than that for the two two-phase systems. It is though worth noting that the AUC metric considers type-I and type-II errors as equally bad and if it is the case for a particular task, the user for sure would prefer the approach with better AUC. However, for the TFBS prediction problem, high false positive rate is always a big challenge, so we prefer to consider the approach with possibly lower AUC since it provides lower false positive rate, rather than a generally superior one.

Second, for HRE prediction by the neural networks, we have the AUC values

that are nearly equal to the products of corresponding sensitivity and specificity. That is, the ROC curves have rectangular shapes, and, therefore, there is no actual trade-off between the type-II and type-I errors. Indeed, neural networks usually converge to particular answers within machine precision for most inputs, so there is little chance for any threshold-mediated balance. If the selection of one particular error type is of greater importance, then it may be reasonable to use another pattern recognition method which allows for more user-defined accuracy trade-off [130], such as those exploiting the statistic models.

We also used the available vertebrate genomes (genome assemblies provided by NCBI) to evaluate the performance of our HRE recognition methods. Average frequencies of prediction of HREs in these genomes are as follows:

- Human genome (*Homo Sapiens*, #36.2) – 1:8.15Kb
- Chimpanzee genome (*Pan Troglodytes*, #2.1) – 1:8.13Kb
- Mouse genome (*Mus Musculus*, #36.1) – 1:7.69Kb
- Rat genome (*Rattus Norvegicus*, #4.1) – 1:7.11Kb
- Cow genome (*Bos Taurus*, #3.1) – 1:6.35Kb
- Dog genome (*Canis Familiaris*, #2.1) – 1:8.43Kb
- Opossum genome (*Monodelphis Domestica*, #2.1) – 1:7.36Kb
- Chicken genome (*Gallus Gallus*, #2.1) – 1:9.81Kb
- Zebrafish genome (*Danio Rerio*, #2.1) – 1:8.95Kb

To conclude the section, we established a novel two-phase neural network architecture, which models HREs by the cascaded combination of the feed-forward and Hopfield neural networks. Using this two-phase architecture we achieved the 92% level of HRE prediction sensitivity, and the random expectation level of 1 prediction per 7.29Kb of neutral DNA and 1:8.15Kb of human genomic DNA. This accuracy rate is the best ever available for predicting steroid HREs. It also outperforms the typical TFBS prediction rates reported before [35].

6.5 Results and Discussion

As stated in the Britannica encyclopedia [161], “neural network is a computer

program that operates in a manner analogous to the natural neural network in the brain”. Artificial neural networks have been found to be extremely useful for machine learning, and they are already widely used in computational biology [162]. Modern kernel-based methods such as support vector machines [163] usually cannot compete with ANNs in terms of flexibility and generality.

In this chapter, we described the design of the novel two-phase neural network model for prediction of symmetrically structured DNA motifs. The model was trained and tested using the set of experimentally validated HRE sequences. The achieved prediction accuracy with sensitivity of 92-98% guarantees that the training set of interest is indeed modeled quite well, and the corresponding selectivity as high as 1 prediction per 7-8 kilo base pairs also outperforms any previously known solution.

Our two-phase approach benefits from the advantages of both of its constituents, namely, the feed-forward and the recurrent neural network, which reflect the different strategies of machine learning. The feed-forward neural network generally provides a flexible tool for modeling almost any dataset of interest, but its flexibility may sometimes result in possibly lower specificity values. This drawback is partially compensated by exploiting the subsequent recurrent neural network. Using the distinct approach for the second phase of HRE recognition we eliminated fortuitous HRE-like though not symmetrically structured findings.

Considering the described idea of structure prediction by the recurrent neural network enhanced with dynamic adaptation, we found that it works well for most of the functional HREs, and provides robust results on the chance of short frame-shifting mutations. However, there exists an unsolved issue that the system may fall into oscillations if a chimerical motif is encountered. For such a motif, one half of the sequence half-site comes from one orientation of the consensus half-site, and the other half – from its reverse form. We met at least two experimentally validated examples of such chimerical HREs identified in the promoter regions of vertebrate genes. Namely, a progesterone response element with the right half-site AGTACT (compare with the HRE consensus TGTCT and its reverse form AGAACA) is known to be involved into regulation of rabbit uteroferrin gene [164], and the same androgen receptor-responsive DNA sequence acts in the promoter area of rat probasin gene [165]. Such cases are marked by our two-phase system as false-negatives and will require more flexible investigation in future.

Even though the cascade of neural networks manages to model the HRE training data with high approximation, there are still some outliers that cannot be covered by the model due to their highly different composition. As it can be inferred from Table 6.1, the training error hardly improves as the complexity of the FFNN increases. Thus, we make the conclusion that at least 2% of collected HREs have nothing in common with the rest of the HRE dataset. Indeed, the misclassified sequences are highly different from the consensus HRE sequence. The non-consensus binding sites for progesterone, androgen and glucocorticoid receptors were found in the promoter regions of a number of genes: rabbit uteroglobin gene [164], chicken lysozyme gene [166], porcine uteroferrin gene [167], pro-opiomelanocortin gene [168], murine c-myc gene [169], late leader of the control region of the human polyomavirus BK [170], gene promoter of two milk protein genes (β -casein and whey acidic protein) [171], human Na/K ATPase $\alpha 1$ gene promoter [172], and mouse sex-limited protein enhancer [173]. The first three genes are progesterone-regulated, the next five are glucocorticoid primary targets, and the last one is associated with androgen activity. If experimental artifacts are not the occasions, a possible explanation could be obtained from the complex nature of protein-DNA interaction, which is beyond the DNA sequence similarity itself. It can probably be due to non-direct but assisted binding like it was described for the glucocorticoid regulatory unit [174], or a secondary structure of DNA [175], or location of surrounding nucleosomes [176], or just non-specific binding [177].

Some particular findings concerning interaction of hormone receptors with their DNA targets are interesting to explore considering their natural background. In particular, a very promising strategy for protein binding site prediction might be established at the level of molecular modeling [178]. Different amino acids (aa) have different binding preferences towards nucleotide bases (bp) addressing the nature and strength of aa-bp interaction [179]. This problem, however, is expected to be extremely complex and requires much more knowledge about protein-DNA interactions *in vivo* than we have today. But nevertheless, we are able to connect our findings with the current knowledge in molecular physics to a certain degree of speculation. One of the observations is the statistically significant preference of the androgen receptor towards A-rich reverse forms of HREs, while the other hormone receptors prefer T-rich direct repeats (refer to Table 6.2).

When the three amino acid sequences of DNA-binding domains for the three hormone receptors (taken from UniProt Knowledgebase 8.6 [180]), namely, the human AR (entry #P10275), human GR (#P04150), and human PR (#P06401), were aligned against each other, five amino acid positions were found to be the same in PR and GR, but different in AR. Two of these aa positions are involved into dimer interface interaction, as confirmed by crystallographic analysis [181]. In both positions, PR and GR has isoleucine (Ile), while AR has threonine (Thr) and phenylalanine (Phe), respectively.

Fig. 6.8 Amino acid sequence of the glucocorticoid receptor DBD. Five amino acid positions which are the same for GR and PR DBDs but different for AR DBD, are marked by arrows with respective amino acid substitutions.

Investigating into the aa-nt interaction data (reported by Luscombe et al. [179]) we found that the dimer-forming amino acid Thr of the AR protein had no hydrogen bonds with the essential guanine of the half-site consensus; instead, it had five bonds with adenine base and four bonds with thymine nucleotide base. This may be the actual reason why AR does not distinguish direct and inverted half-sites. Moreover, the Phe amino acid has nearly twice as much van-der-Waals contacts with adenine as the Ile (22 vs. 12). Of course, these inferences must be verified experimentally but they indeed can provide a starting basis to the possible explanation of the AR, but not PR and GR, being tolerant to the inversed HRE half-sites as well.

The two main disadvantages of our two-phase neural method is the requirement for a large training set for proper model learning, and the presence of the iterative process for HRE structure prediction. The latter results in long training times and

makes the highly accurate approach at least comparable to the simpler ones like those exploiting the statistical models, in terms of usefulness. In the case of real genomic sequences, which are billions of base pairs now, the process HRE recognition already takes prohibitively long time. However, high-performance solutions, and, in particular, the application-specific hardware acceleration methods may be useful to boost the running time of the proposed neural architecture in order not to compromise its superior accuracy when predicting symmetrically structured DNA motifs

Chapter 7

Achieving Convergence of the Neural Network with Hybrid Computing

In the previous chapter, we proposed the two-phase neural network architecture for recognizing HREs in genomic DNA. This approach demonstrates the HRE prediction accuracy that outperforms any previously known solution, but its execution time appears to be prohibitively long for practical use in case of vertebrate genomes. To solve this problem, we propose the FPGA architecture to speed up the computations of the neural network model for HRE recognition. In Section 7.1, we estimate the computational complexity and running time for the two-phase neural system proposed in Chapter 6, which motivates the development of a high-performance computing solution. In Section 7.2, we describe the distributed hardware architecture for the dynamically adaptable Hopfield neural model designed for the dimeric structure prediction. In Section 7.3, we introduce the hybrid CPU-FPGA neural system which is aimed to recognize HREs in DNA and ensure affordable computational resources and superior prediction accuracy. In Section 7.4, we consider the application of the developed neural model for prediction of steroid hormone primary target genes with the example of progesterone-regulated human genes. Finally, in Section 7.5, we summarize and discuss the results and performance of the developed neural system.

7.1 Computational Complexity of Neural Network

Computational complexity of the two-phase neural system designed for recognition of HREs emerges from complexities of its constituents, namely, the feed-forward neural network designed for HRE sequence modeling, and the Hopfield neural classifier for dimeric structure prediction. Though the achieved prediction accuracy is sound, the system's demands for computational resources require a special consideration.

Computational complexity of the feed-forward neural network mainly pertains to the training process, while the operation of the trained neural network is

straightforward and therefore fast. Specifically, the training complexity is $O(n \times L \times I \times h)$ where n is the number of training sequences, L is the length of the pattern of interest, I the amount of learning cycles, and h is the size of the hidden layer. Additionally, FFNN testing requires only $O(n \times L \times h)$ operations.

On the contrary, for the case of HNN, it is its recurrent operation process that gives the largest impact on the computational complexity. The complexity of one iteration for the exact-match Hopfield neural architecture is $O(L^2)$, where L is the length of the HRE pattern. For the dynamically adaptable matching employed by the DP-HNN, additional preprocessing of $O(L^2)$ complexity is required. Additionally, as the iterative process of HNN convergence is repeated for each input, an important characteristic of the recurrent architecture is its convergence speed. For both EM-HNN and DP-HNN, we estimated the median number of iterations required for reaching a stable state (medians were used instead of means because absence of oscillations during the convergence process could not be guaranteed while the mean value would be heavily affected by a single instability). For the EM-HNN, the average median number of iterations is 480, and for the DP-HNN it is 400 for all experimentally validated HREs.

The neural networks designed for HRE recognition and the auxiliary applications for data preprocessing were implemented as C/C++ applications for Win32 with the cygwin gcc compiler involved for executable file generation. Using the IBM four-processor X260-server with 3.17GHz CPUs and 3.25Gb of RAM controlled by Win2003 Server operation system, it took about 8,000 seconds to process 1 Megabyte of DNA text with the FFNN followed by the EM-HNN model, and 10,000 seconds for the system with the second recognition stage performed by the DP-HNN. However, if the trained FFNN operated solely for HRE recognition, it required only 30 seconds to finish the analysis of 1Mb of genomic data. Additionally, the process of learning the FFNN with 40 hidden neurons took nearly 300 seconds using the dataset of 7,000 input vectors.

We also tested a parallel implementation of the described neural architecture on a four-processor CPU. In that implementation, the neurons of the HNN were processed by four parallel threads. We got the running time of 2,400 seconds for 1Mb DNA screening using the FFNN followed by the EM-HNN, and 2,800 seconds when the FFNN was followed by the DP-HNN.

The conclusion is that the bottleneck for the two-phase system is indeed the HNN

phase. While the feed-forward neural network is trained once and forever, and then its operation is a straightforward pass through the sequence of its layers, the recurrent model requires hundreds of iterations for each input vector. The order of its execution time does not allow using the software implementation of HNNs for genome-wide HRE recognition purposes. Furthermore, the dynamic adaptation results in 10 to 20 percent increase of calculation time.

Large-scale reconfigurable computing provides a wide range of possibilities to exploit the full potential of recurrent ANNs. Although some experts believe that it is difficult to expect a custom neuro-computer to operate significantly faster than the best general-purpose computer [182], it is still worth using hardware implementations of ANNs as co-processing parts of the application-specific neuro-systems. Resulting high-performance hybrid machines thus may outperform any conventional solution.

7.2 Dynamically Adaptable Hopfield Neural Network on FPGA

In most existing applications, ANNs are implemented in software. However, as the complexity of the network grows, network training and operation time usually becomes prohibitively long. Concerning the HNN designed for prediction of HRE dimeric structure, it takes hours to process megabytes of DNA using a very powerful PC. However, a hardware-accelerated implementation of the recurrent neural network looks like a promising solution, particularly with reference to the conclusion that the FPGA implementation of the multi-stage Markov model designed for HRE recognition (refer to Chapter 5) confirmed the viability of the approach.

When dealing with hardware implementations of neural networks, mapping of the ANN operating mechanism to hardware is a crucial issue. Designing such a mapping is also a big challenge. There exist several approaches for implementing and training ANNs on a chip, while choice of a particular design and learning strategy usually depends on the problem specification. Some, though rare, tasks may require high flexibility [183]. For others, performance of a given architecture is of concern [184–186].

Back-propagation method is generally used for learning both feed-forward and recurrent neural networks. It is though particularly difficult to implement large-scale ANNs with the gradient-descent learning because of the complexity of the mechanism that derives the gradient. As an alternative, the simultaneous perturbation method was

introduced by Maeda and Tada [187]. It avoided the difficulties associated with the back-propagation learning by substituting the gradient with a simple error estimation function. Additionally, for the recurrent neural networks used for unsupervised pattern recognition, extra steps are required to specifically design the network model, but, fortunately, most of the time- and area-consuming steps of on-chip learning can be avoided [188]. A comprehensive review of recent progress in hardware implementation of neural networks using the FPGA technology is given by Zhu and Sutton [189].

Based on the digital circuit design technology, we mapped the Hopfield neural network designed for HRE structure prediction on an FPGA chip, and then used that chip as a co-processing unit for the two-phase neural system. The FPGA co-processor communicates with the host PC via the local bus. We used Verilog HDL for FPGA circuit design. The design was simulated and synthesized using the Xilinx ISE and Xilinx XST software. The Alpha Data Virtex-4LX160 PCI chip with 135 168 logic elements (8M gates) and 288x18kbit RAM blocks was used. Alpha Data SDK 4.6.0 API was used for programming the communication layer between the front-end application and the on-chip HNN unit.

In our implementation, the CPU-FPGA system operates as follows. The C++ application reads 15bp-long DNA sequences from a text file, which contains either putative HREs predicted by the FFNN module, or functional HREs from the collected dataset. The application then executes the DNA encoding module Σ and sends the resulting vector of 60 bit values, representing the one-hot encoded DNA sequence, to the configured FPGA board. It also obtains the output from the board, and passes that data to the decision making module.

In the following subsections, we describe the technical aspects of our HNN circuit design addressing the methods used for efficient data transfer and computations.

7.2.1 Calculation units

For implementation of numerical values on FPGA we adapt a fixed point representation in two's complement notation. The values of neuron weights and outputs are the 32 bit numbers with one sign bit, two integer bits and 29 fractional bits. This representation is sufficient to describe operation of the HNN with acceptable precision.

In order not to exhaust the limited number of logic gates, we use the 32bit×32bit multiplications for generation of the synaptic inputs to the neurons, which operate in

the form of finite state machines. Each of these machines regulates the operation of two dedicated hardware 18bit×18bit multiplier units. Henceforth, the 32bit arithmetic operations will be referred as an adder and multiplier, respectively.

Control and verification units

The *control unit* is used for serial processing of the input vector by the HNN unit. For that purpose, the control unit dispatches control signals to all other units. In particular, the control unit is responsible for sequential processing of data flow.

The operation of the control unit is shown as a state flow diagram in Fig. 7.1.A. At the initial state, the network input is set up. Then, the recurrent alignment procedure is performed as described in Section 6.4. Resulting alignments of the two half-sites are then processed in parallel by the two groups of the *neuron units*. Each group contains four physical neuron units and performs computations for four neurons (NeuronQ, or neuron quartet, in the figure) at a time. Each neuron quartet corresponds to a single nucleotide encoded using one-hot notation for the DNA alphabet, so the depth of the cycle is equal to the length of the half-site of interest. For HREs, the half-site consensus consists of six nucleotides.

Fig. 7.1 A. State flow of the dynamically programmed Hopfield neural system. B. Verification unit controls the HNN recurrence.

The *verification unit* is shown in Fig. 7.1.B. The unit contains an array of 20×60 32-bit registers, an oscillation detection module, and a counter of iterations. Current

DP-HNN output, which is an array of sixty 32-bit fixed-point numbers, is placed into one column of twenty registers in a cyclic order, so that the set of registers always stores the network outputs from the twenty most recent iterations. The oscillation calculation module thus computes the relative oscillation of the HNN output during the twenty consecutive iterations. The HNN iterations are directed to stop by that module when the total absolute deviation is less than $1/2^{10} = 1/1024 \sim 0.1\%$ of the output value for all neurons, so we conclude that a stable state is reached. The counter of iterations signals a stop when the maximum number of iterations is exceeded. After testing the network operation, we set the maximum number of iterations to 10,000 which is a very conservative estimation. Even without dynamic programming, which provides a better match and therefore faster convergence, we found it to be enough for the HNN to converge with majority of the inputs unless a specific pattern caused substantial oscillations in the network.

The stop signal produced by the verification module causes the whole DP-HNN unit to finish the processing of the current input. Current output is then sent to the PCI bus to the awaiting front-end application, which conveys to it the user interface or to the decision making scheme.

Matching unit

The recurrent alignment procedure is performed by the *matching unit* shown in Fig. 7.2. It obtains the input vector, performs the procedure of alignment for its half-sites, and returns an array of weight index shifts which are used for further processing of the vector by proper HNN neurons. An input vector is preprocessed by the Pre-Processing Unit (PPU on the figure), which defines the indices of maximum elements for each consecutive four elements (thus defining a particular nucleotide base encoded by them), and then only these indices are used.

Inside the matching unit, the two identical half-site processing modules perform the recurrent alignment procedure for the two consensus HRE half-sites, each for one of its orientation, namely, the direct (left) or the inverted (right part of the figure). In order to decrease the number of registers for sequence representation, consensus vectors are stored in the chip memory and never change during the network operation. Each half-site processing module uses two 6-bit addressed RAM sections. One RAM section stores the alignment scoring matrix M , and the other one stores the trace-back array D for reconstructing the resulting optimal alignment.

A finite state machine regulates the succession of scoring matrix calculations. Though it is possible to implement alignment score matrix in linear space [131], we are dealing with the iterative process so it is more important to minimize the time latency. That is, our implementation of the recurrent alignment procedure consumes the space quadratic to the length of the half-site since it is affordable for the case of relatively short HRE motifs. However, such an implementation permits simultaneous calculation of two matrix elements at a time: the cell (i,j) and its symmetric cell (j,i) , which are contoured with bold in Fig. 7.2. Additionally, the initial matrix values, which are stored in its first row and first column, can be filled simultaneously if we use the gap penalties of degrees of two, thus avoiding costly multiplications.

Fig. 7.2 The matching unit is responsible for the recurrent alignment procedure.

The total latency of the half-site alignment procedure for an input of length n is, therefore, $2 + \frac{n(n+1)}{2} + 2n - 1$. Specifically, two sequential operations are required for the input preprocessing, and they are parallelized with one operation of initialization of the scoring matrices, $n(n+1)/2$ operations are required to compute both alignment scoring matrices of size $n \times n$, and $2n - 2$ operations are needed to select the maximum values from their last columns and last rows. Finally, at most $2n$ operations are necessary to reconstruct the resulting alignment using the trace-back matrix D. For

the HRE half-site of length 6, the latency of the alignment module is at most 52 calculation cycles.

Outputs of each half-site processing module are the best alignments of the half-sites and their scores. Then, a multiplexer controlled by a selector picks out the alignment with the highest score, and the downstream processing module returns the best alignment in the form of index shifts for neuron weights.

7.2.2 Neuron and memory units

The main part of the on-chip HNN consists of the neuron units which are connected to the memory unit. The memory unit stores the neuron weights, as well as current and initial input and output vectors.

The neuron unit is implemented using several types of calculations, namely, the adder, the multiplier, and the register. Inside the neuron unit, two consecutive elements of the input vector and the appropriate weight values are multiplied at a time using two 32bit×32bit multiplier modules operating in parallel. Thus, the latency of the neuron unit operation is half of the input vector length. The unit's addressing scheme required for communication with the memory unit uses the index shifts for neuron weights returned by the matching unit. The computed weighted inputs for the neurons are stored in the memory registers, and then summed up into the synaptic input of the neuron. The schematic RTL structure of the neuron unit is shown in Fig. 6.3.

The output of the neuron is computed from its synaptic input using its activation function. In our implementation, we use a linear approximation of the sigmoid function

$$F_0(x) = \frac{1 - e^{-x}}{1 + e^{-x}}.$$

The approximation is used in order to avoid calculations of actual

sigmoid, which requires series of multiplications and at least one division. Instead, we use a combination of linear curves with slopes of 1/2, 1/4, and 1/8. Thus, the resource-consuming multiplication and division operations are replaced by “cheap” register shift operations.

Four instances of the neuron unit are used in parallel in each of the two parts of the on-chip HNN, so the total amount of neuron units involved is eight. Each part of HNN processes the input half-site. Inside each part, after a neuron quartet is calculated (the end of calculation is reported to the control unit by a handshake signal), it is replaced by the next four, of total twenty-four for each of the HRE half-sites.

The memory stores the weight values for the neurons and the outputs of the

neurons. It is also used for storing the twenty recent HNN outputs used for oscillation measurements, the initial input, weight index shifts, and the iteration counts. These data are read or written according to the commands of the control unit. Since we have 48 neurons in the HNN, there are 48×48 32-bit weight values. Therefore, we need the 12-bit addressing scheme. Verification unit stores 20×60 32-bit values and requires 11-bit addressing scheme.

Fig. 7.3 The HNN neuron unit. The pruned part of the figure is the section of distributed memory that stores the input vector.

7.2.3 Operation of the DP-HNN

The operation of the on-chip DP-HNN system is mainly divided into three functional units: the matching unit, the actual HNN unit, and the verification unit. Fig. 7.4 shows the configuration of the DP-HNN from the point of view of digital data processing.

The top-level control is performed by the counter of iterations. It defines when the counter of neurons should be reset to zero, whether a vector has to be put into the system as its initial input, which register of the verification unit contains the oldest output, and whether it is time to stop because the maximum allowed number of iterations is exceeded. The counter of neurons is reset before the HNN iteration starts, and then it regulates the succession of neuron quartets processing, as well as decides

which values of weights must be selected from memory using the corresponding weight index shifts.

Fig. 7.4 Configuration and control of the DP-HNN on FPGA.

The operation of the HNN is implemented by eight identical physical neuron units grouped into two sections, and the memory. The neuron unit performs sum-of-products operations for calculation of neuron synaptic inputs at its Run mode. Two groups of four input elements are processed in series. That is, eight physical neuron units emulate 48 neurons as required by our HNN-based HRE recognition model. The weight values, inputs, and outputs are read from or written to the distributed memory.

Figure 7.5 shows the data flow in the DP-HNN FPGA system. The results of simulation were generated by the trial version of Xilinx ModelSim III tool (<http://xilinx.com/mxe3>).

Summary of the ensued implementation for the ADM Virtex-4 chip is as follows:

Logic elements:	101,696 of 135,168 (75%)
RAM:	960Kbit of 5,184Kbit (19%)
I/O pins:	101 of 960 (11%)
DSP slices:	48 of 96 (50%).

Chapter 7. The Hybrid Adaptive System

138

Fig. 7.5 Data flow in the DP-HNN FPGA system.

The described HNN implementation was developed, simulated, and compiled using Xilinx ISE 9.1 hardware design environment.

7.3 Hybrid Adaptive System for HRE Recognition

The proposed two-phase neural architecture for HRE recognition consists of the trained feed-forward neural network used for modeling of the HRE sequences, and the Hopfield neural network used for validating those which are the dimeric motifs. The two-phase architecture is implemented as a CPU-FPGA hybrid neural system. In such a system, the Hopfield neural network unit implemented on the FPGA is used as a co-processor for the whole system operation.

Fig. 7.6 Hardware-accelerated hybrid adaptive system for HRE recognition.

As shown in Fig. 7.6, the putative HREs returned by the first neural network are further processed by the on-chip HNN module. The HNN output is then passed to the

decision making module that eventually returns the binary answer for the putative HRE predicted at the FFNN stage. The sequences marked as HREs by the whole system are then stored into a text file ready for further processing and analysis.

In order to evaluate the speed up gained using the FPGA-based hardware acceleration, we developed a software version of the same Hopfield classifier (for both its exact-match and dynamically adaptable versions). A software application was tested on the four-processor IBM server. For comparison with purely software solutions, we implemented the software versions of the HNNs with both one thread and four threads being processed in parallel by four CPUs. The FPGA on-chip clock frequency was set to 100 MHz.

Table 7.1 confirms a nearly 50X speed-up of the hardware-accelerated version versus the single-threaded software implementation, and a 10X speedup versus the high-performance software implementation. Additionally, the dynamic adaptation causes nearly 10-20% increase of calculation time. It is interesting to note that, as it was observed, software implementations may require less operation cycles to converge even though the models implemented in hardware and software are essentially the same. The explanation is that the software implementation exploits 64-bit floating point numbers, while the FPGA solution uses fixed point number representation with 32 bits and thus introduces more imprecision into the calculations.

Table 7.1 Computational performance of the hardware-accelerated system implemented using the Virtex-4 FPGA and the four-processor IBM PC.

HNN	Implementation	Processing time (sec)	
		HRE database	1Mb of DNA
EM-HNN	V-4 FPGA	0.5	250
	C++ 4 threads	2	2,400
	application 1 thread	8	8,000
	V-4 FPGA	1	300
DP-HNN	C++ 4 threads	3	2,800
	application 1 thread	10	10,000

For conclusion to the section, the described hardware-accelerated solution for HRE recognition by the two-phase neural architecture results in an order of magnitude of acceleration and, thus, affordable execution time. That is, the described hardware-accelerated system can be used for genome-wide HRE recognition in evolutionary studies, functional genomics, and molecular endocrinology.

7.4 Prediction of Progesterone Primary Target Genes

To reconstruct the first layer of the hormone-regulated gene expression network, it is essential to define hormone primary target genes. Reliable information is available for only a few experimentally validated primary target genes, mostly because current experimental methods can hardly provide high-throughput data.

7.4.1 The progesterone primary target genes

The steroid hormone progesterone is of special interest to our group. Progesterone is involved in the regulation of a considerable number of vertebrate genes. However, the distinction between direct and indirect targets of progesterone action inside cells is largely yet to be made. By separating the regulatory events which are primary for the molecular effects of progesterone from the secondary consequences of progesterone action, it may be possible to define the determinants of response to progesterone in normal and malignant cells.

Several progesterone primary target genes have been reported before. In the work by Ahola *et al.* [190], *GPR30* gene encoding the G-protein-coupled receptor 30 was reported to be a progesterone primary target gene. Treatment with is a protein synthesis inhibitor cycloheximide did not prevent progestin-induced upregulation of *GPR30* and had no effect on *GPR30* mRNA levels after 24 hours.

Another progesterone primary target gene is Human Insulin Receptor Substrate-2 (*IRS2*). Vassen *et al.* [191] identified that partial inhibition of protein synthesis by cycloheximide did not inhibit induction of human *IRS2* mRNA by progesterone.

As reported by Tsuchiya *et al.* [192], mouse gene *EP2* (prostaglandin E receptor) is a progesterone-regulated gene, and, as it was shown by mutational analysis, its upstream promoter area contains three regions involved into basal promoter activity in the presence of PR. Therefore, the authors conclude that *mEP2* gene is a progesterone primary target gene.

In the work by Kester *et al.* [193], an improved differential display technique was implemented, and its output was the list of ten progesterone target genes: *TSC22* (TSC22 domain family, member 1), *CD9* antigen (p24), *ATP1A1* (encoding for ATPase, Na⁺/K⁺ transporting, alpha 1 polypeptide), *DSP* (desmoplakin), *CD59* antigen p18-20, *FKBP5* (FK506 binding protein 5), *ANXA6* (annexin A6), and three unknown genes.

Mouse Irg1 (Immune-Responsive Gene 1) is also a progesterone target gene. It plays a critical role during blastocyst implantation [194]. The gene HDLG5/KIAA0583, which encodes a MAGUK-family protein, is a primary progesterone target gene in breast cancer cells [195]. Additionally, the progesterone receptor activates its promoter activity in human endometrial stromal cells [196].

In a recent work by Leo *et al.* [197], extensive screening of potential progesterone primary target human genes was performed. As a result, using the translation inhibition experiment, the authors received a validated list of 26 primary target human genes. Most of those were declared as progesterone-regulated genes for the first time.

7.4.2 Method for prediction of primary target genes

To predict primary target genes *in silico*, we accomplished the HRE recognition procedure using the hardware-accelerated system for the set of upstream promoter regions of all available vertebrate genes. Additionally, we considered the validated progesterone-regulated genes (microarray-based data kindly provided by Dr. Valerie Lin, with reference to Leo *et al.* [197]). Nucleotide sequences of gene promoter regions were retrieved from the Genbank NCBI database which contained annotated genomes for the vertebrate species including human (*Homo Sapiens* #36.2), mouse (*Mus musculus* #36.1), rat (*Rattus Norvegicus*, #4.1), dog (*Canis familiaris* #2.1), chimpanzee (*Pan troglodytes* #2.1), chicken (*Gallus Gallus* #2.1). Though a number of other genomes are also available now, to the moment of research they were quite new and therefore poorly annotated.

For each of the used genomes, the genes' localization, positions of their transcription start sites (TSSs), and the exon-intron structures are given in the Genbank assembly (Fig. 7.7). Detailed annotation of every chromosome contig is followed by its nucleotide sequence, so that one can extract the DNA sequences of promoter regions for all annotated genes. For the purpose of primary target gene prediction, we use the upstream promoter regions which were the regions [-2000;+10] relative to the TSSs. This area is believed to contain a functional promoter even if it is relatively distant. A small part of the first exon is also included, because the annotation databases may contain approximate or ambiguous TSS locations.

```

ref_chr22.gbk
LOCUS       NT_011519               3661581 bp    DNA    linear    CON 29-AUG-2006
DEFINITION  Homo sapiens chromosome 22 genomic contig, reference assembly.
ACCESSION   NT_011519 NT_011518
VERSION     NT_011519.10  GI:29806588
KEYWORDS
SOURCE      -
ORGANISM    Homo sapiens
            Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
            Mammalia; Eutheria; Primates; Catarrhini; Hominidae; Homo.
REFERENCE   1 (bases 1 to 3661581)
AUTHORS     International Human Genome Sequencing Consortium.
TITLE       Finishing the euchromatic sequence of the human genome
JOURNAL     Nature 431 (7011), 931-945 (2004)
PUBMED      15496913
COMMENT     GENOME ANNOTATION REFSEQ: Features on this sequence have been
            produced for build 36 version 2 of the NCBI's genome annotation
            (see documentation).
            On or before Mar 1, 2006 this sequence version replaced
            gi:10879004, gi:17484914.
            The DNA sequence is part of the fourth release of the finished
            human reference genome. It was assembled from individual clone
            sequences by the Human Genome Sequencing Consortium in consultation
            with NCBI staff.
            COMPLETENESS: not full length.
FEATURES             Location/Qualifiers
     source           1..3661581
                     /organism="Homo sapiens"
                     /mol_type="genomic DNA"
                     /db_xref="taxon:9606"
                     /chromosome="22"
     gene             23010..23791
                     /gene="ABCD1P4"
                     /note="synonym: ALD22Q11"
                     /pseudo
                     /db_xref="GeneID:26957"
                     /db_xref="HGNC:65"
     exon             23020..23096
                     /gene="ABCD1P4"
     mRNA             3638545..3638548,3638674..3638818,
                     3638928..3639025,3639314..3639405,3639509..3639678,
                     3640254..3640424,3640990..3641025,3641293..3641335)
                     /gene="LOC440795"
                     /product="similar to carbonic anhydrase 15"
                     /note="Derived by automated computational analysis using
                     gene prediction method: GNOMON. Supporting evidence
                     includes similarity to: 2 Proteins"
                     /pseudo
                     /transcript_id="XR_016957.1"
                     /db_xref="GI:113429305"
                     /db_xref="GeneID:440795"
     ORIGIN
1   gatcatcatc gaatggagtt gaatggaatt atcaaagaat ggaatccagt ggtatcatca
61  tcaaatggaa ccgaatggaa tcatcaaatg gactcaaatg gaatcattga atagattcga
121 atggaatcat catcgaatga atcgaatgg aaaaattgaa tggactcgaa tggaaaccatc
181 attgaatgga aaccaaagga atcatcatct aatgaaatga aatggaatca tagaatggac
241 acaaatggaa tcatcatcga atggtattga atggaatcat cgaaaagaat cgacggaatc
301 atgatcaaat ggactcgaat ggaatcatca tggaaatgga tcaaatggaa tcatcaatga
361 atggaatcca atggtatcat catcaaatgg aaccgaaagg aatcatcaag tggactcaaa
421 tggaaatcatc gaatggaatc gaatggaatc atcatcgaat ggaatcgaat ggaacatcgy
481 aatggaatcg aatggaaaca tcattgaaat gaatcatcat caaacggaat ctatttgaat
541 cctcattgga tagaatcgaa tggaaatcatc aaatggaata gaaaggaatc atcatcgaat
601 gcaatcgaat agaatcatca aatgaaaagg aacggaatca tcatcgaatg gaatcgaatg
661 gaatcatcaa cgaatggaat cgaatggaat catcgtctaa tggaaatcaa tggaaatcatc
721 aacgaatgga atagcatgga atcatcgaat ggaatctcat ggcacatca tcacatggaa
781 ccgaatggaa tcatcatgga gtgtaatcta atggaatcat cattgaatgg aatccaatgg
841 gatcactgaa ttgaatggaa tgatcatcga atggaatcaa agggaaatcat caaatgggat
901 agaaggggat catagaatgg aatcgaatgg aatcatcgaa tggattcaaa tggaaatcacc
961 atcaaatgga aaagaatgga atcatcaaaa ggactcgaat ggaatcatca aggaatagaa
1021 tcaaatggaa taattgaata gacacgaatg gaatcatcat ttaatggaat caaatggaat
1081 catcgaatgg actcaaatgg aatcatcatt gaatggaatc gaatggaatc atcgaatgac
1141 attgaataga ataataatg aatggaatct taaggaaata tcgaatggac tcgaatggaa
1201 taatcgaatg gactcgagtg gaatcatcat cgaatggaat cgaatggaat catcaaatgg
1261 actcaaatgg aatcatcatt gaatggaatg gaatgaaatc atcgaatgga ctcgagtgga
1321 atcatcatcg aatggaatgg aatggaatca tcgaatggac tcaaatggaa tcatcatcga

```

Fig. 7.7 Genbank file with human chromosome 22 assembly.

The NCBI HomoloGene [198] is a system for automated detection of homologs among the annotated genes of several completely sequenced eukaryotic genomes. For

the purpose of progesterone primary target genes prediction, HomoloGene is used for extraction of vertebrate orthologous genes. Among all 380 PR-responsive human genes [197], most are accompanied with orthologs (according to the HomoloGene build #41.1) with exception of 13 genes. Among these 13, all the human genes are in fact predicted genes supposed to encode for hypothetical proteins or open reading frames.

With a list of HREs recognized in the promoter areas of progesterone-responsive human genes, we apply a qualitative significance calculation scheme to estimate the probability of each gene to be a progesterone primary target. Our method is based on two widely accepted assumptions: functional response elements should be preserved during evolution and thus conserved among different species [13], and a complex of adjacent response elements is more likely to be functional than a single binding site. Regarding the second assumption, it was demonstrated by Tsai et al. during the series of EMSA experiments, that 80% of DNA primers with two adjacent response elements for glucocorticoid and progesterone receptors formed protein-DNA complexes, while single HREs did so for only 15-20% of primers [199].

Accordingly, we propose the following primary target scoring scheme where we incorporate the reliability of each predicted HRE in the gene promoter area:

- (i) Score of the predicted HRE itself equals to 1.
- (ii) If a similar HRE is found in a similar position within the promoter area of an ortholog, the HRE score is increased by one. We call the positions similar if the difference of the HRE distances from TSSs is up to 200 bp, and a similar HRE is the one with matching of more than 50% of the response element's length and the same or reverse dimeric structure.
- (iii) If a similar HRE is not found in a similar position within a promoter of an ortholog, the HRE score is decreased by 0.5. The mismatch penalty is lower than the match reward because in our approach, the HRE prediction sensitivity is much lower than the specificity. Moreover, not all genomes are fully annotated, especially the newly reported ones, such as that of dog, chicken, and chimpanzee. Therefore, the absence of an object is less significant than its presence.
- (iv) If two or more HREs are located within 100bp distance, then we increase the HRE score by one for each of them.

- (v) The scores for all HREs predicted in the gene promoter region are then summed up, and the result is the primary target score of the gene promoter, further referred to as *PT*. If the gene promoter does not contain any HREs, its score is set to -55.

Among all the scored genes, only those with *PT* above 3 are marked as possible progesterone primary target genes. The threshold of 3 may be tuned. However, in most cases, the score value of at least 3 implies either presence of many HREs per promoter thus giving a higher probability of having at least one to be functional, or one or two HREs with high evolutionary conservation.

The scheme effectively reduces false positive level because the evolutionally conserved and densely clustered HREs result in higher *PT* scores for a gene. Response elements of these genes are more likely to be functional and, in turn, the prediction of primary target genes is more accurate and reliable.

It is necessary to note, however, that for the described scoring scheme, only one species, namely, the *Homo sapiens* is used as the center of scoring, since its HREs are evaluated for similarity to the others. The other species are not compared to each other and their pairwise similarity is not taken into account.

7.4.3 Prediction results

The set of gene promoter regions extracted from genomes of vertebrate species was used for HRE prediction by the system. The average number of HREs recognized in promoter areas of 380 PR-responsive genes was 0.85, while for the whole set of human genes this value was as low as 0.27 HREs per gene promoter.

The histogram of *PT* score values over the set of human PR-responsive genes is shown in Table 7.2 and Fig. 7.8. Among them, 147 genes with no HREs in their promoter regions are characterized with the default *PT* score value of -55 and are not included. The shape of the histogram of *PT* score values in Fig. 7.8 resembles Gaussian, but the curve has peaks at integer numbers and falls at fractional numbers. This is mainly due to the scoring scheme intrinsic properties: it has higher probability for assigning an integer number.

Table 7.2 Primary target score values of human PR-responsive genes.

Summary score of the gene promoter	11	9	8	8	6.5	6	5.5	5	4.5	4	3.5	3
Number of genes	1	1	1	1	1	4	2	2	5	5	9	17

Summary score of the gene promoter	2.5	2	1.5	1	0.5	0	-0.5	-1	-1.5	-2	-2.5	-3
Number of genes	6	19	25	18	31	33	12	18	10	5	0	5

Fig. 7.8 Histogram of primary target score values of human PR-responsive genes.

The top 32 PR-responsive human genes with the *PT* score above 3 are summarized in Table 7.3.

Table 7.3 Human genes with the highest scores of being progesterone primary targets.

	Gene name	Gene ID	Gene description	Number of predicted HREs	Summary score of the gene promoter
1	<i>MMP1</i>	4312	matrix metalloproteinase 1 (interstitial collagenase)	3	11
2	<i>ATBF1</i>	463	AT-binding transcription factor 1	4	9
3	<i>NGFB</i>	4803	nerve growth factor, beta polypeptide	3	8
4	<i>NAV3</i>	89795	neuron navigator 3	3	7.5
5	<i>Hist2h2aa</i>	8337	histone 2, H2aa	3	6.5
	<i>RIS1</i>	25907	Ras-induced senescence 1	3	6
	<i>LAPTM5</i>	7805	lysosomal associated multi-spanning membrane protein 5	3	6
6-9	<i>CSF2</i>	1437	colony stimulating factor 2 (granulocyte-macrophage)	4	6
	<i>CMAH</i>	8418	cytidine monophosphate-N-acetylneuraminic acid hydroxylase (CMP-N-acetylneuraminate monooxygenase)	6	6
10-11	<i>UCP2</i>	7351	cell cycle controller CDC2	2	5.5
	<i>IER3</i>	8870	immediate early response 3	2	5.5
12-13	<i>BMP4</i>	652	bone morphogenetic protein 4	3	5
	<i>IRS1</i>	3667	insulin receptor substrate 1	5	5
	<i>ITGA3</i>	3675	integrin, alpha 3 (antigen CD49C, alpha 3 subunit of VLA-3 receptor)	3	4.5
14-18	<i>EXOSC9</i>	5393	exosome component 9	3	4.5
	<i>FOSL1</i>	8061	FOS-like antigen 1	2	4.5
	<i>CALD1</i>	800	caldesmon 1	3	4.5
	<i>PHLDA1</i>	22822	pleckstrin homology-like domain, family A, member 1	3	4.5
	<i>ARL7</i>	10123	ADP-ribosylation factor-like 4C	2	4
	<i>ADAM8</i>	101	a disintegrin and metalloproteinase domain 8	4	4
19-23	<i>EXT1</i>	2131	exostoses (multiple) 1	2	4
	<i>SDPR</i>	8436	serum deprivation response (phosphatidylserine binding protein)	2	4
	<i>MT1K</i>	4499	metallothionein 1K	2	4
24-32	<i>PLA2G4A</i>	5321	phospholipase A2, group IVA (cytosolic, calcium-dependent)	4	3.5
	<i>RASA1</i>	5921	RAS p21 protein activator (GTPase activating protein) 1	1	3.5
	<i>LIFR</i>	3977	leukemia inhibitory factor receptor	1	3.5
	<i>SPRY2</i>	10253	sprouty homolog 2 (Drosophila)	1	3.5

<i>SLCO4A1</i>	28231	solute carrier organic anion transporter family, member 4A1	3	3.5
<i>FLJ22794</i>	63901	FLJ22794 protein	2	3.5
<i>IL1RAPL1</i>	11141	interleukin 1 receptor accessory protein-like 1	2	3.5
<i>KYNU</i>	8942	kynureninase (L-kynurenine hydrolase)	1	3.5
<i>TM7SF2</i>	7108	transmembrane 7 superfamily member 2	1	3.5

The highest *PT* score was found for human *MMP1* gene encoding for matrix metalloproteinase 1. Its promoter contains three predicted HREs and two of them are adjacent. The alignment of the promoter area of human *MMP1* gene and promoters of its five orthologs is shown in Fig. 7.9. The alignment was performed by the ClustalW software with its default parameters [200]. Box frames indicate locations of two adjacent HREs. In the figure, hs = *Homo Sapiens*, pt = *Pan Troglodytes*, cf = *Canis Familiaris*, mm = *Mus Musculus*, rn = *Rattus Norvegicus*, gg = *Gallus Gallus*.

Fig. 7.9 HREs in the promoter areas of human *MMP1* gene and its orthologs.

Our findings are also well-related to the previous results reported by molecular biologists. In particular, the steroid hormone progesterone has previously been known to down-regulate the human *MMP1* gene expression [201]. The second significant PR-responsive gene *NGRF* has also been reported to be progesterone-regulated [202]. Human genes *NAV3*, *CSF2*, *IER3*, *BMP4*, *FOSL1*, *PHLDA1*, *IRS1*, *ARL7* were demonstrated to be regulated by progesterone in presence of a translation inhibitor, and hence, are the progesterone primary genes too [197]. Interestingly, all of these genes are known to be down-regulated by progesterone, while none of the vertebrate genes known to be up-regulated is recognized by our model. That makes us to suspect a special mechanism for up-regulation of transcription by progesterone receptor which may be carried out by non-specific binding to DNA or interaction with other

regulatory proteins without interaction with the gene promoter region itself.

According to the description of our scoring algorithm, it is a center-star-like approach since the human genes are compared to their orthologs, but the orthologs are not compared to each other. To estimate the stability of the approach, we performed the same scoring algorithm using mouse, rat, dog, and chimpanzee genomes as the ‘center-stars’ of the scoring scheme, in turn. The genes with *PT* score value of at least 3 from each of these species were used to render a Venn diagram shown in Fig. 7.10.

Fig. 7.10 Venn diagrams for the hormone primary target genes in vertebrates. Notation: a) human (HS), mouse (MM) and chimpanzee (PT); b) human (HS), rat (RN) and dog (CF);

In these diagrams, human and chimpanzee graphs are overlapping significantly, but despite this, there exist some differences. Considering the actual high similarity of chimpanzee and human genomes [203], the predicted primary target chimpanzee genes which are not primary target genes for human are likely false positives.

Other vertebrate species also demonstrate high but not full similarity to human. It is important to note, though, that only human PR-responsive genes are used as the basis of the current model. Thus, smaller numbers of primary target genes for other species can be explained, to some extent, by the predefined selection of human primary targets.

It is well-known that the mechanisms of transcription regulation, being rather complicated themselves, become even more intricate when considered including their

induced responses which may be quite remote. However, with more experimental information becoming available, it is promising to look further and investigate into induced effects of the first level of hormonal regulation.

In the available list of PR-regulated genes, there are at least eight genes whose product proteins are known to be involved into transcriptional regulation. One such gene, namely, *FOSL1* is proved to be a progesterone primary target by both our computational method and wet-lab experiments with translation inhibition [197].

The Fos gene family consists of four members: *FOS*, *FOSB*, *FOSL1*, and *FOSL2* [204]. These genes encode leucine zipper proteins that can dimerize with proteins of the *JUN* family, thereby forming the transcription factor complex *AP-1*. The Fos proteins have been implicated as regulators of cell proliferation, differentiation, and transformation. Those are the processes for which the progesterone activity is extremely important as well. At the same time, the interleukin-8 gene, which is known to be progesterone regulated, is also regulated by a Fos transcription factor [205]. Therefore, it might be interesting to look at the hypothetical pathway of gene expression regulation performed as follows: progesterone → human *FOSL1* gene → Fos transcription factor → regulation of *IL-8*. Such a sequential gene expression analysis involving more extensive biological knowledge and more advanced computational technologies might eventually lead to establishing the entire hormone-regulated gene expression network.

7.5 Summary and Discussion

During the last decades, research in molecular biology and medicine provided the scientific community with a huge amount of data through sequencing, genome-wide annotations, gene expression profiling, and proteomics projects. The genetic databases keep growing exponentially; in particular, the GenBank of NCBI nearly doubles in size each year, and the whole-genome sequences of many different organisms has already been completed or will be completed in the nearest few years. Another breakthrough of the post-genomic era has been reported very recently: now the first genomic portrait of an individual, rather than the genome assembly which we used to have after the completion of the human genome project, is available. The first “sequenced” person was the “father” of the first assembled human genome Dr. J. Craig Venter [206].

Having fast and cheap DNA sequencing techniques in hand, we may expect numerous new genome sequences to become available very soon. That is, the problem of computational processing of biological data is rapidly becoming extremely time- and resource-consuming, especially in the perspective of individual genomes' availability. The development of optimization algorithms and high-performance technologies becomes as important as invention of novel theoretical approaches and gaining the new knowledge. A promising technology that harmonizes the power of application-specific hardware circuits and availability of designing via relatively simple development tools, thus making the approach accessible to a wide variety of scientists, is the FPGA technology.

In this chapter, we presented a novel hybrid neural network-based HRE recognition system, and tested its applicability for prediction of steroid hormone primary target genes. As demonstrated, exploiting of the FPGA technology results in the achieved speed-up of nearly an order of magnitude in comparison with the high-performance solution operated solely in software, thus providing us with a powerful pattern recognition tool that successfully handles the problem of accurate and specific prediction of symmetrically structured DNA motifs.

The main challenge of our FPGA design is finding the right balance between bit and node parallelism in order to reach the best overall performance and keep the applicability to the chosen domain given the implementation constraints. In the current implementation, we process the input vector using eight physical neuron units, and each unit involves four embedded multipliers. However, we may trade the time efficiency for more complex activation rules in order to improve the numerical precision. In particular, we use 29 fractional bits for number representation, and thus obtain the imprecision of up to -10^{-8} per HNN iteration. For the case of steroid HREs, that error is not critical because the number of iterations does not exceed the level of 500 for most inputs. However, if we now consider more complex motif patterns, it may result in significantly worse HNN convergence. In turn, the worse convergence will lead to processing of thousands of recurrent HNN cycles, and the resulting error due to number imprecision will not be negligible any more.

Another interesting aspect of the FPGA design is the proper chip area management. As an illustration of the importance of this issue, in Table 7.4 we give the results of FPGA implementation of the EM-HNN which was not specifically described in this thesis. It was mostly similar to the described DP-HNN, with the only

difference being the absence of the matching unit, and also ten neurons units instead of eight were placed onto the chip (the EM-HNN architecture was described by the authors just before the dynamic adaptation technique was developed [156]).

Table 7.4 Summary of two different FPGA implementations of the EM-HNN.

Neuron latency (activation cycles)	Activation ALUs per neuron	Mult18x18 per neuron unit	Summary of ensued FPGA design			
			Logic elements	RAM	I/O pins	DSP slices
60	1	2	90,576 of 135,168 (67%)	732Kbit of 5,184Kbit (14%)	98 of 960 (10%)	36 of 96 (38%)
30	2	4	140,932 of 135,168 (104%)	984Kbit of 5,184Kbit (19%)	101 of 960 (11%)	77 of 96 (80%)

In the referred implementation, each neuron unit contained only one ALU with two embedded multipliers, and the area usage was close to 70%. When we tried to decrease the latency of computations and involved the second ALU into the neuron module, the synthesizer returned the results summarized in the table. As shown, the decrease of neuron latency from 60 to 30 activation cycles by doubling the number of involved multipliers caused overuse of the FPGA board area due to the lack of logic gates. Thus, the HNN architecture with twice lower time latency could not be implemented on the particular FPGA chip solely because of limited area resources.

In conclusion to the chapter, we designed a two-phase neural model with partial hardware acceleration for recognizing symmetrically structured DNA motifs in genomic sequences, and evaluated its performance using the collection of experimentally validated HREs. Though the proposed design is currently implemented reflecting the specific HRE structure, it is possible to consider an extension of the approach to the entire family of nuclear receptor response elements, and possibly involve other symmetric, dimeric, and repeated motif structures.

Part IV

Concluding Remarks

Chapter 8

Discussion and Conclusion

In this Chapter, we discuss the proposed computational models and the experimental results, summarize the main contributions of the thesis research, and conclude with the possible ways for future work. In Section 8.1 we summarize the work done. In Section 8.2, we highlight what has been achieved and assess the performance and applicability of the invented algorithms. In Section 8.3, we propose the directions for further research, ranging from development of more advanced bioinformatics algorithms to incorporation of more molecular biology knowledge.

8.1 Summary on the Thesis Research Contributions

A comprehensive survey of computational methods was conducted with the aim to understand the state-of-the-arts technologies in the relevant research, especially, the niche area of computational modeling of hormone-regulated gene expression was found to be challenging. The biomedical problem itself, however, is of great importance. Development of efficient tools for computational recognition of hormone response elements and hormone primary target genes is necessary.

For training the computational models, we first collected a representative set of HREs from our in-house experiments and biomedical literature. Experimentally reported homogeneity of steroid HREs was further validated using the HRE dataset that had been created through extensive search for functional response elements for the three steroid hormone receptors of interest, namely, the androgen, glucocorticoid, and progesterone receptors.

The statistic position-transition model was established with reference to the multi-component HRE structure. Based on the validated observation that HREs carry statistically significant di-nucleotide preservation, higher-order weight matrix models were evaluated together with the conventional first-order PWM. The PWM-based position frequency model was further enhanced by the nucleotide transition model implemented using the specifically designed multi-stage cascade Markov model. The ensemble of the position and transition models demonstrated significantly improved sensitivity and specificity of HRE prediction.

While the position-transition model implies additivity of impacts of the single or di-nucleotide residuals into the process of protein-DNA interaction, the neural network considers each input sequence as an integral object and, therefore, provides a better solution for the case of a non-monomeric motif structure. We designed and tested the feed-forward neural network as a method for separating the HRE sequences from neural DNA. The result shows that the multi-layer perceptron learned using the large and properly balanced training set can successfully maintain highly sensitive HRE recognition. Additionally, for increment of the selectivity in HRE prediction, we designed a Hopfield neural network for describing HREs from a different point of view, and incorporated the two methods into the two-phase prediction system.

Pioneering work for predicting dimeric structures of DNA motifs was carried out. Since the hormone receptor dimerization is known to be essential for successful protein-DNA interaction, the idea of dimeric structure modeling by the specifically designed Hopfield neural network allowed us to eliminate a large portion of false positive HRE predictions. To efficiently reduce the number of false positives when predicting HRE motifs, the candidate HREs were firstly selected using a highly sensitive recognition method, which was the feed-forward neural network, and then each candidate HRE was tested for holding a dimeric structure. The proposed two-phase neural system outperforms any previously known solution for recognition of nuclear receptor response elements. We also developed a novel HNN extension technology based on dynamic programming for more flexible sequence-structure matching. The approach was tested for steroid HREs only, but it can possibly be applied for other symmetrically structured DNA motifs.

Finally, we employed reconfigurable computing technology for hardware acceleration of the cascade Markov model and the recurrent neural network in order to resolve the bottleneck problem in computation of the multi-stage HRE prediction process.

A number of examples of gene regulatory sequences with pairs of regulators, as well as multiple copies of the same binding signals with fixed or preferential spacing between site occurrences, are observed in nature. Thus, it is likely that the next generation of tools for recognition of regulatory elements will be focused on predicting site combinations rather than distinct transcription factor binding sites. The proposed high-performance methods for recognition of symmetrically structured DNA motifs described by the example of hormone response elements open the door to the direction.

8.2 Critical Analysis and Assessment of the Proposed Methods

Transcription factor binding site recognition, though is conceptually a well understood task, has some very challenging constraints. The typically short length of TFBSs poses a big problem as it significantly increases their chances to occur randomly. Hence, high false positive rate has always been a limiting factor for precise TFBS recognition. The possible solution for eliminating the excessive false positive predictions is to design multiple-feature recognition schemes reflecting the specific characteristics of a particular binding site or a family of those.

In this work, we designed and evaluated novel computational methods for prediction of symmetrically structured HRE motifs, and developed the new hardware-accelerated implementation for the most time-consuming steps of the multiple-feature HRE prediction frameworks.

8.2.1 Recognition of HREs using statistic models and neural networks

Receptors for sex steroid hormones are transcription factors that bind to specific regions in DNA in order to facilitate regulation of gene expression. While the biological and molecular basis of interactions between the receptor molecules and their DNA response elements is not fully understood, the sequence pattern of the HREs is known to be partially and non-trivially conserved. To date, there has been a lack of effective bioinformatics tools for recognition of partially symmetric DNA motifs and for HRE recognition as well.

In Parts II & III of the thesis, we introduced different models for recognizing the TFBSs with partially symmetric structure, and tested them by the example of steroid HREs. These models consider different features of the DNA pattern subjected to modeling. In particular, position weight matrices calculate the impacts of the mono- or oligo-nucleotide constituents of the HRE sequence to protein-DNA interaction independently and thus result in the additive position-specific motif representation. The cascade Markov model refers to the state-transition pattern of the HRE sequence. Generally speaking, these two models consider the input sequence with reference to its successive nucleotide composition using relative frequencies of its elements; that is, we indicate the resulting position-transition model as a nucleotide-specific, or additive, or statistic. On the other hand, neural network models consider their input sequences as integral objects. Thus, the neural-network approach is indicated as object-specific.

Unsurprisingly, the statistic and neural network models demonstrated different performance during training and testing. The statistic position-transition model, in addition to its intrinsic constraints for sequence modeling, is based on the known HRE consensus with rigid arrangement of the half-sites and the spacer, and is limited to recognizing motifs with a matching composition only. On the contrary, the proposed two-phase neural architecture allows for certain flexibility of HRE modeling, so that the limitation of the exact matching between the putative HRE and the known response element prototype is overcome. However, each of the models has its own advantages, and none could be thrown away as if completely outperformed.

In Table 8.1, we briefly summarize the HRE prediction accuracy and the worst-case computation time of the proposed statistic and neural network models. The notation in the table is as follows: the abbreviations PWM1 and PWM2 denote the first- and the second-order position weight matrix models, respectively (refer to Chapter 3), MM denotes the cascade Markov model (refer to Chapter 4), the prefix HA- denotes the hardware-accelerated implementation of the following algorithm (refer to Chapters 5 and 7), FFNN denotes the feed-forward neural network with 40 hidden neurons (refer to Section 6.1), EM-HNN denotes the Hopfield neural network with exact match convergence paradigm (refer to Section 6.2), and DP-HNN denotes the HNN with dynamic alignment described in Section 6.3. The 95% confidence intervals for the prediction sensitivity and specificity were calculated using a trial version of the MedCalc statistics tool (<http://www.medcalc.be>). The computation time was measured using 1Mb of randomly generated DNA sequence. For testing the multiple-feature prediction frameworks, each input sequence was processed by all algorithms, in order to measure the worst possible performance.

For comparison, accuracies of four existing tools developed for HRE prediction are also given in the table. Dragon ERE Finder [24] is developed for ERE prediction only and, though exploits a position frequency model, demonstrates excellent prediction capability. The significant limitation of the tool, however, is its limited reference set: the model is trained and tested using only 30 functional EREs and 30 non-ERE sequences. NHR-Scan [106] is the tool designed for recognition of all nuclear HREs without limitation to steroid HREs. Due to diversity of nuclear receptors, the tool has limited accuracy even though a promising idea of using an HMM specifically designed for recognition of dimeric motifs has been proposed. P-Match [85] is the tool that uses TRANSFAC [82] PWMs. SayaMatcher [207] uses

PWMs provided by JASPAR database of PWMs [83] but also considers genomic context of found HREs. Sensitivity and specificity values are cited from the authors' websites (for P-Match) or respective papers.

Table 8.1 Performance of the developed and existing methods of HRE recognition.

Prediction method	Runtime, sec/Mb	Sensitivity, %	Specificity, kbp ⁻¹	AUC
PWM1	40	86±4	1380±210	0.95±0.03
PWM2	40	86±4	1025±155	0.94±0.03
PWMs	80	76±4	3180±350	0.96±0.03
Markov model	50	84±4	990±150	0.92±0.03
HA-MM	8	83±4	1000±150	0.92±0.03
PWMs+MM	120	83±4	3550±420	0.96±0.03
FFNN	20	98±4	5840±780	0.98±0.04
FFNN-EM-HNN	2400	92±4	7440±1180	0.96±0.03
FFNN- HA-EM-HNN	300	91±4	7290±1130	0.96±0.03
FFNN-DP-HNN	2800	96±4	7280±1230	0.97±0.03
FFNN-HA-DP-HNN	350	96±4	7080±1210	0.97±0.03
Dragon ERE Finder [24]	na	83	13300	na
NHR-Scan [106]	na	~80	~1000	na
P-Match [85]	na	100	352	na
SayaMatcher [207]	na	na	~5700	na

Fig. 8.1 HRE prediction ROC curves for the statistic and neural network models.

Figure 8.1 shows the ROC curves for the typical case of HRE prediction using a statistical model (by the example of PWM1) and a neural network model (by the example of FFNN). The graphs demonstrate one of the (dis)advantages for the nucleotide-specific vs. object-specific approaches to the problem of TFBS modeling. Though the neural network provides a better prediction rate, its ROC curve is almost a step function and thus does not allow for any trade-off between the prediction sensitivity and specificity. On the contrary, the ROC curve for the statistic model is much smoother, though located farther from the 100% accuracy point.

Though the neural networks obviously outperform the statistic models, like position weight matrices and Markov models, in terms of prediction accuracy, the simpler methods have their own advantages and thus should not be discarded by default. Not only the prediction accuracy, but the cost of development and execution, design requirements and limitations, and the scope of applicability are the important characteristics of a computational algorithm. Below, we summarize the advantages and disadvantages of the pattern recognition methods involved for TFBS modeling which have been inferred from the tests using the example HRE motifs.

Position weight matrix

+: popular, easy to design, tolerant to the size of training set, fast execution, easily interpretable representation of sequence profiles, flexible user-defined prediction trade-off;

-: low space for application-specific adaptation, low prediction selectivity.

Markov model:

+: easy to design, fast execution, allows for designing complex application-specific models for non-trivial patterns, flexible user-defined prediction trade-off;

-: low prediction selectivity.

Feed-forward neural network:

+: fast execution, high sensitivity, acceptable prediction selectivity;

-: more complicated design and training, computationally expensive learning, requires large training sets for non-over-fitted learning, little space for user-defined trade-off.

Hopfield neural network:

+: the most accurate prediction of structured motifs, robust and flexible structure prediction with dynamic adaptation;

-: complicated design, prohibitively long execution time, no user-defined trade-

off, depends on proper selection of initial conditions.

To conclude the comparison, the statistic models, like PWMs and Markov models, should be preferred for TFBS prediction in either very large or very small DNA sequences. In particular, for the case of very long sequences, like complete genomes, the exact positions of each binding site are usually not that important; instead, the trends of TFBS frequencies and their density distributions are considered, as it is done for evolutionary studies [208]. Dealing with very small DNA sequences, it might be important to recognize as many putative binding sites as possible, even if those are burdened with a large portion of false positives. For a short sequence, the predictions are usually not abundant, and each can be further validated *in vitro* [209].

However, in real life, we usually deal with mid-long DNA sequences like gene promoter regions. For the problem of *in silico* modeling of gene expression regulation, location of each functional regulatory element is important and large amount of false positive signals may lead to inconsistent inferences. That is, the proposed multiple-feature frameworks for accurate TFBS recognition are effective as precursors to experimental validation and also provide grounds for biologically meaningful conclusions.

8.2.2 High-performance machine learning based on reconfigurable computing technology

Applications that push the limits of computer technology come from a variety of fields, including financial analysis, data mining, medical imaging, and scientific computation, each with widely diverse computational requirements. What these high-performance computing applications all have in common, however, is the critical need for the highest possible speeds of program execution. In the last few years, there has been a movement towards clusters, or so-called grid computing technologies, as an answer to the rising performance demands. Now, however, the cluster systems based solely on general-purpose processors are running out of room to grow. Adding field-programmable gate array co-processors to these systems can boost application performance while simultaneously reducing time and power consumption and the total cost of ownership.

For the problem of HRE modeling, we employed the hardware acceleration technology for the two pattern recognition methods: the multi-stage cascade Markov model, and the dynamically adaptable Hopfield neural network. The on-chip models

were further used as co-processors for enhanced HRE recognition by the multiple-feature frameworks. For the case of DP-HNN, we achieved a tenfold speedup comparing with the multithreaded software implementation run on a very powerful PC, and nearly 50X speedup comparing to the single-threaded implementation for the general-purpose CPU. Thus, the proposed FPGA implementations of the algorithms benefit from both parallelization and application-specific logic interconnections, while the designed architectures allows for easy updates facilitating the resources vs. latency trade-off. Resolving the bottlenecks with the hardware acceleration technologies, we make the proposed HRE prediction tools efficient for genome-wide analysis without compromising the prediction accuracy.

A more general conclusion is that hardware acceleration widens the limits of bioinformatics approaches to recognition of motifs in biological sequences. The last but not the least note is that though bioinformatics is sometimes referred to as “*in silico* biology”, with involvement of reconfigurable computing methods it becomes truly *in silico*.

8.3 Recommendations for Future Work

The task of modeling the hormone response elements consists of three main and generally independent stages. The first stage is collecting a training set of HREs. The second stage is establishing a reliable prediction model and developing a learning scheme using the training collection and relevant biological knowledge. Finally, the third stage is implementing high-performance technologies that would allow achieving the best possible accuracy values in affordable time.

The first step looks like to be developed at an appropriate level to date, though also might be improved by separation of the training data into classes. The reason is that though the steroid HREs are usually considered as the same for androgen, progesterone, and glucocorticoid receptors, they were demonstrated to hold some hormone-specific features as well [210]. However, the other two modeling steps should be paid more detailed attention.

8.3.1 Beyond binding sites

While numerous HREs can be identified using various *in silico* prediction methods, the question remains as follows: are these HREs are actually functioning as

HREs *in vivo* given that most regulatory regions are packaged into nucleosomes and are thus inaccessible without chromatin remodeling? The futility theorem states that nearly 100% of the TFBSs predicted using computer-based tools will show no functionality *in vivo* [25]. Additionally, Horie-Inoue *et al.* [209] stated that out of the 565 perfectly palindromic HREs that they found using computational prediction, only 26 sequences were within 10kb near the TSS, and out of those 26 sequences only 14 showed functional activity when validated. Therefore, the factors, which influence HRE binding activity other than sequence itself, must be considered.

Phylogeny.

The idea of evolutionary development of existing species from one ancestor is a basis of comparative genomics [211]. The main assumption for phylogenetic analysis is that mutations in functional regions of genomes are more critical for normal development, and therefore are exposed to stronger selective pressure, than those in genomic regions with no specific function. In particular, transcription factor binding sites located in well conserved areas are considered as probably selected through the process of evolution and, thus, more likely functional.

After completion of sequencing the whole genomes for several vertebrate species, it has become possible to use the results of inter-species comparisons in practice for tuning the methods of computational recognition of regulatory elements [212]. Moreover, it was already shown, that phylogenetic filter allowed to reduce the level of false positives by orders of magnitude in exchange for a modest sensitivity decrease [213]. To date, the most promising and well suited tool for phylogenetic alignment is mLAGAN [214]. It allows for multiple sequence global progressive alignment of long genomics regions. When accompanied with a tool for visualizing the results of alignment (e.g. Sockeye [215]), mLAGAN provides the near-term solution.

Attempting to improve the specificity of TFBS prediction, different research groups developed prediction tools that combined PWMs or other statistic modeling methods with phylogenetic analysis. ConSite [213] and rVista [216] are the examples. Both programs were reported to eliminate up to 90% of false predictions while retaining 70-80% of experimentally validated sequences [13]. However, Mustonen and Lassig [217] argued that those methods applied a conservation criteria that was somewhat heuristic from an evolutionary point of view and did not provide a

consistent measure of statistical significance of predicted binding sites.

A different algorithm, proposed by Elnitski et al. [218], used patterns of the nucleotide distribution among a number of species in a region around the putative binding site, and classified the region as a coding, regulatory, or neutral, according to its *regulatory potential*. The idea of the regulatory potential came from the observation that coding regions tended to vary at the third positions within codons, and also tended to have indels that were multiples of three. Per contra, regulatory sequences tended to have more frequent indels and variations that occurred in distinct blocks and were separated by segments of high similarity. The method, which was implemented as a hidden Markov model, represents an approach that is likely to become increasingly important as more genome sequences and more sophisticated alignment algorithms become available.

However, we should remember that alignment-based phylogenetic footprinting methods are relevant for orthologous genes from species with appropriate evolutionary divergence. Pairwise alignment and comparison of promoters from closely related species, such as human–chimpanzee, generally provides little benefit because the sequences closely resemble each other, whereas the promoters from widely divergent species (primate–fish) can show no detectable similarity [81]. The rate of evolutionary events in promoters is also different for genes within the same organism. Therefore, in some cases, it is productive to compare sequence pairs from more diverged species.

Context-based criteria.

Several assumptions underlie the most popular methods for TFBS prediction. One of them is that every transcription factor binds to its target DNA independently while regulating gene transcription. That is, we assume that binding is not influenced by the content of adjoining sequences and the proximity of other proteins. This assumption does not always hold, and a strong limitation follows immediately: the proposed methods are unable to specifically distinguish between the binding sites that have a functional role *in vivo* and the sites with no function.

In fact, in higher eukaryotes, gene regulation is often mediated by cooperative interactions between transcription factors that bind to clusters of sites within *cis*-regulatory modules (CRM) [44]. A gene can have multiple CRMs in its surrounding non-coding sequence. Typically, several different regulatory proteins bind within an enhancer, and each transcription factor can bind to multiple sites. Regulatory proteins

may differ in the structure of DNA binding, activation, oligomerization, and other domains. Along with structural differences, functional properties of the transcription factors, and hence their specific contribution to the transcription regulation, may vary significantly. Co-operative action of transcription factors within regulatory modules results in a highly specific pattern of gene transcription that cannot be provided by these factors involved separately.

Regulatory clusters can possibly be modeled by computational algorithms and used for further improvement of TFBS recognition. Although this direction of regulatory network design is very recent, several research groups have already proposed relatively sophisticated algorithms for recognition of TFBS clusters. The software tool TRANSCompel [219] provided an opportunity for modeling the composite regulatory elements which consisted of TFBSs predicted using the TRANSFAC database and the MATCH prediction tool. Another algorithmic approach for recognition of *cis*-regulatory modules was proposed by Rajewsky *et al.* [220]. Accompanied by the most impressive results, the algorithm exploits sliding a window along a gene promoter sequence, and evaluates the window's content p-value – the probability that the given density of regulatory signals appeared by chance. The algorithm RISO proposed by Carvalho *et al.* [221] uses a data-structure called “box-link” to store the information about the conserved regions in order to specifically predict functional CRMs. Similar research works with other ideas for CRM modeling were reviewed in [13, 81].

The context of gene regulatory elements is undoubtedly crucial for their functionality and for functionality of HREs as well [222, 223]. This issue promises to be intensively developed in the nearest future, with more accurate and fast methods of motif recognition appearing. With reference to this assumption, it is important to note that the proposed two-phase neural architecture, currently designed for modeling and recognition of symmetrically structured motifs, can be extended for the purpose of recognition of paired or repeated clustered motifs in *cis*-regulatory regions.

In conclusion, the exact mechanism for sex steroid hormone regulating gene expression is yet to be elucidated. The bottom line is that gene expression regulation by different steroid hormones is a very complex process, in particular, due to the complex nature of interactions between receptor proteins and their response elements in DNA. A number of *in silico* methods were developed for recognizing functional

HRE sequences, others can be used for enhancement the methods beyond HREs, and each method always has its advantages and disadvantages. Utilizing several methods in an ensemble usually improves the accuracy of prediction of functional HREs in comparison with standalone methods. Additionally, even focusing on the process of initiation of transcription, other mechanisms of regulation along the gene expression pathway should not be neglected. And of course, experimental methods are still required to validate the HREs and candidate primary target genes identified by computational approaches.

8.3.2 Expanding the computational limits

In recent years, reconfigurable computing platforms based on the FPGA technology have emerged as viable alternatives for many types of high-performance computing applications. The opportunities provided by these platforms include the rapid creation of custom hardware, simplified updates in the field, and reduction or elimination of other application-specific chips in many categories of electronic products. As FPGAs have also grown in logic capacity, their ability to host high-performance algorithms and applications has grown respectively. Now, in addition to general purpose processors and common processor peripherals, a PC platform can include one or more customized, highly parallel software/hardware accelerators.

Considering the FPGA-based pattern recognition methods developed for HRE modeling, we expect at least two possible ways for improving the performance. The first is the clock management, and the second is the enhanced data transportation.

Generation and distribution of clock signals has a significant impact to the performance and power dissipation of the system. The clock signal can have both spatial and temporal variations, which lead to performance degradation and/or circuit malfunction. In particular, the clock skew (which is a spatial variation in arrival time of a clock transition) may appear in the case of relatively large circuit areas occupied by the design. Other unavoidable reasons, namely, manufacturing variations, interconnect variations, vertical and horizontal dimension variations, and environmental variations, also exist. Due to these reasons, clock frequency and, therefore, overall computational performance of the FPGA system is usually limited by the developer, in order to guarantee the stability of the on-chip calculations. For example, for the proposed HRE recognition architectures, we limited the FPGA clock frequency to 100 MHz while the Virtex-4 technology allowed increasing it to 500

MHz, though at the cost of high risk of calculation bugs. Thus, distributed fault-tolerant clock management system is expected to increase the performance of the entire on-chip model [224].

Considering the data transport issue, the limited speed of the connection between the host PC and the co-processing FPGA may be a problem. In particular, the PCI bus for the FPGA board used in our project can have frequency of 33 MHz or 66 MHz only, comparing to the frequencies of modern CPUs which are nearly two orders of magnitude higher. Low bus speed may sometimes become a bottleneck, especially when calculations cannot be performed solely on-chip. Though this is not the case for the Hopfield neural network, such a problem may occur for high-throughput techniques with straight-forward data processing, like multi-component Markov models. Using HyperTransport technology for more efficient data transportation and faster host-chip communication, we may increase the performance of the hybrid systems. HyperTransport, formerly known as Lightning Data Transport, is a bidirectional serial/parallel high-bandwidth low-latency point-to-point link that was first introduced in 2001 [225]. It is expected to provide faster communication than the conventional PCI buses or JTAG interfaces may do, for the applications which involve highly intensive data flow.

Bibliography

1. Conneely OM (2001) Perspective: Female Steroid Hormone Action. *Endocrinology*, 142(6): 2194-2199.
2. Eddy EM, Washburn TF, Bunch DO, Goulding EH, Gladen BC, Lubahn DB, and Korach KS (1996) Targeted disruption of the estrogen receptor gene in male mice causes alteration of spermatogenesis and infertility. *Endocrinology*, 137(11): 4796-4805.
3. Pelzer T, Shamim A, Wolfges S, Schumann M, and Neyses L (1997) Modulation of cardiac hypertrophy by estrogens. *Adv Exp Med Biol*, 432: 83-89.
4. Cutolo M, Sulli A, Capellino S, Villaggio B, Montagna P, Serio B, and Straub RH (2004) Sex hormones influence on the immune system: basic and clinical aspects in autoimmunity. *Lupus*, 13(9): 635-638.
5. Maggi A, Ciana P, Belcredito S, and Vegeto E (2004) Estrogens in the nervous system: mechanisms and nonreproductive functions. *Annu Rev Physiol*, 66: 291-313.
6. Kearns AE and Khosla S (2004) Potential anabolic effects of androgens on bone. *Mayo Clin Proc*, 79(4S): 14-18.
7. Ko YJ and Balk SP (2004) Targeting Steroid Hormone Receptor Pathways in the Treatment of Hormone Dependent Cancers. *Curr Pharm Biotechnol*, 5(5): 459-470.
8. Stanford JL, Stephenson RA, Coyle LM, Cerhan J, Correa R, Eley JW, Gilliland F, Hankey B, Kolonel LN, Kosary C et al. (1999) *Prostate Cancer Trends 1973-1995. SEER Program, National Cancer Institute*. NIH Pub, Bethesda, MD.
9. Cook JD and Walker CL (2004) Treatment strategies for uterine leiomyoma: the role of hormonal modulation. *Semin Reprod Med*, 22(2): 105-111.
10. Lin CY, Strom A, Vega VB, Kong SL, Yeo AL, Thomsen JS, Chan WC, Doray B et al. (2004) Discovery of estrogen receptor alpha target genes and response elements in breast tumor cells. *Genome Biol*, 5(9): R66.
11. Kumar S, Saradhi M, Chaturvedi NK, and Tyagi RK (2006) Intracellular localization and nucleocytoplasmic trafficking of steroid receptors: an overview. *Mol Cell Endocrinol*, 246(1-2): 147-156.
12. Alberts B, Bray D, Lewis J, Raff M, Roberts K, and Watson J (1994) Intercellular signalling. *Molecular Biology of the Cell*. Garland Publishing, New York.
13. Wasserman WW and Sandelin A (2004) Applied bioinformatics for the identification of regulatory elements. *Nat Rev Genet*, 5(4): 276-287.

14. Fickett JW and Wasserman WW (2000) Discovery and modeling of transcriptional regulatory regions. *Curr Opin Biotechnol*, 11(1): 19-24.
15. Hu J, Li B, and Kihara D (2005) Limitations and potentials of current motif discovery algorithms. *Nucleic Acids Res*, 33(15): 4899-4913.
16. Kuncheva LI and Whitaker CJ (2003) Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy. *Machine Learning*, 51(2): 181-207.
17. Wang R, Gao LQ, Yang S, and Liu YC (2005) An edge detection method by combining fuzzy logic and neural network. *Proc of Intl Conference on Machine Learning and Cybernetics*, 7: 4539-4543.
18. Szekely N, Toth N, Lasztovicza L, and Pataki B (2004) Combining Methods For Mass Detection In Mammograms. *Proc of Biennial International EURASIP Conference BIOSIGNAL*, 281-283.
19. Cheng V and Li CH (2006) Personalized Spam Filtering with Semi-supervised Classifier Ensemble. *Proc of IEEE/WIC/ACM Intl Conference on Web Intelligence*, 195-201.
20. Oza NC and Tumer K (2008) Classifier ensembles: Select real-world applications. *Information Fusion*, 9(1): 4-20.
21. Khorasanizadeh S and Rastinejad F (2001) Nuclear-receptor interactions on DNA-response elements. *Trends Biochem Sci*, 26(6): 384-390.
22. Benos PV, Bulyk ML, and Stormo GD (2002) Additivity in protein-DNA interactions: how good an approximation is it? *Nucleic Acids Res*, 30(20): 4442-4451.
23. Stormo GD (2000) DNA binding sites: representation and discovery. *Bioinformatics*, 16(1): 16-23.
24. Bajic VB, Tan SL, Chong A, Tang S, Strom A, Gustafsson JA, Lin CY, and Liu ET (2003) Dragon ERE Finder version 2: A tool for accurate detection and analysis of estrogen response elements in vertebrate genomes. *Nucleic Acids Res*, 31(13): 3605-3607.
25. Vavouri T and Elgar G (2005) Prediction of cis-regulatory elements using binding site matrices--the successes, the failures and the reasons for both. *Curr Opin Genet Dev*, 15(4): 395-402.
26. Bulyk ML, Johnson PL, and Church GM (2002) Nucleotides of transcription factor binding sites exert interdependent effects on the binding affinities of transcription factors. *Nucleic Acids Res*, 30(5): 1255-1261.
27. Walters JP, Meng X, Chaudhary V, Oliver T, Yeow LY, Schmidt B, Nathan D, and Landman J (2007) MPI-HMMER-Boost: Distributed FPGA Acceleration. *J VLSI Sig Process S*, 48(3): 223-238.

28. Chen C and Rajapakse JC (2007) Grid-Enabled BLASTZ: Application to Comparative Genomics. *J VLSI Sig Process S*, 48(3): 301-309.
29. Diamond T (2007) FPOAs Meet the Challenges of H.264 Encoding of High Definition Video. *Journal of FPGA and Structured ASICs*, July 31, 2007.
30. Rao N (2007) FPGAs and Ethernet. *Journal of FPGA and Structured ASICs*, July 10, 2007.
31. Nedjah N (2005) Embedded cryptographic hardware : design & security. Nova Science Publishers.
32. Mason M and O'Neill K (2005) FPGA Reliability in Space-Flight and Automotive Applications . *Journal of FPGA and Structured ASICs*, September 6, 2005.
33. Hawkins J and Boden M (2005) The Applicability of Recurrent Neural Networks for Biological Sequence Analysis . *IEEE/ACM Trans Comput Biol Bioinform*, 2(3): 243-253.
34. Workman CT and Stormo GD (2000) ANN-Spec: a method for discovering transcription factor binding sites with improved specificity. *Pac Symp Biocomput*, 467-478.
35. Rahmann S, Muller T, and Vingron M (2003) On the power of profiles for transcription factor binding site detection. *Stat Appl Genet Mol Biol*, 2(1): Article 7.
36. Dayhoff M (1965) Atlas of Protein Sequence and Structure. National Biomedical Research Foundation.
37. Berman HM, Westbrook J, Feng Z, Gilliland G, Bhat TN, Weissig H, Shindyalov IN, and Bourne PE (2000) The Protein Data Bank. *Nucleic Acids Res*, 28(1): 235-242.
38. Sanger F, Coulson AR, Hong GF, Hill DF, and Petersen GB (1982) Nucleotide sequence of bacteriophage lambda DNA. *J Mol Biol*, 162(4): 729-773.
39. Fleischmann RD, Adams MD, White O, Clayton RA, Kirkness EF, Kerlavage AR, Bult CJ, Tomb JF *et al.* (1995) Whole-genome random sequencing and assembly of *Haemophilus influenzae* Rd. *Science*, 269(5223): 496-512.
40. Higgins DG and Sharp PM (1988) CLUSTAL: a package for performing multiple sequence alignment on a microcomputer. *Gene*, 73(1): 237-244.
41. Lipman DJ and Pearson WR (1985) Rapid and sensitive protein similarity searches. *Science*, 227(4693): 1435-1441.
42. International Human Genome Sequencing Consortium (2001) Initial sequencing and analysis of the human genome. *Nature*, 409(860): 921.

43. Venter JC, Adams MD, Myers EW, Li PW, Mural RJ, Sutton GG, et al. (2001) The sequence of the human genome. *Science*, 291(5507): 1304-1351.
44. Alberts B, Bray D, Lewis J, Raff M, Roberts K, and Watson J (1994) Control of Gene Expression. *Molecular Biology of the Cell*. Garland Publishing, New York.
45. Jacob F and Monod J (1961) Genetic regulatory mechanisms in the synthesis of proteins. *J Mol Biol*, 3: 318-356.
46. Kadonaga JT (1998) Eukaryotic transcription: an interlaced network of transcription factors and chromatin-modifying machines. *Cell*, 92(3): 307-313.
47. Carey M and Smale ST (2000) Transcriptional Regulation in Eukaryotes: Concepts, Strategies and Techniques. Cold Spring Harbor, NY.
48. Chlebowski RT, Hendrix SL, Langer RD, Stefanick ML, Gass M, Lane D, Rodabough RJ, Gilligan MA et al. (2003) Influence of estrogen plus progestin on breast cancer and mammography in healthy postmenopausal women: the Women's Health Initiative Randomized Trial. *JAMA*, 289(24): 3243-3253.
49. Li CI, Malone KE, Porter PL, Weiss NS, Tang MT, Cushing-Haugen KL, and Daling JR (2003) Relationship between long durations and different regimens of hormone therapy and risk of breast cancer. *JAMA*, 289(24): 3254-3263.
50. Jensen EV, Cheng G, Palmieri C, Saji S, Makela S, Van Noorden S, Wahlstrom T, Warner M et al. (2001) Estrogen receptors and proliferation markers in primary and recurrent breast cancer. *Proc Natl Acad Sci U S A*, 98(26): 15197-15202.
51. De Vivo I, Huggins GS, Hankinson SE, Lescault PJ, Boezen M, Colditz GA, and Hunter DJ (2002) A functional polymorphism in the promoter of the progesterone receptor gene associated with endometrial cancer risk. *Proc Natl Acad Sci U S A*, 99(19): 12263-12268.
52. General Conclusions on Sex Hormones. *IARC Summary and Evaluation 1979*, 21: 131.
53. Tsai MJ and O'Malley BW (1994) Molecular mechanisms of action of steroid/thyroid receptor superfamily members. *Annu Rev Biochem*, 63: 451-486.
54. Kershaw EE and Flier JS (2008) Adipose tissue as an endocrine organ. *J Clin Endocrinol Metab*, 89(6): 2548-2556.
55. Evans RM (1988) The steroid and thyroid hormone receptor superfamily. *Science*, 240(4854): 889-895.
56. Michalik L, Auwerx J, Berger JP, Chatterjee VK, Glass CK, Gonzalez FJ, Grimaldi PA, Kadowaki T et al. (2006) International Union of Pharmacology. LXI. Peroxisome proliferator-activated receptors. *Pharmacol Rev*, 58(4): 726-741.

57. Willy PJ, Umesono K, Ong ES, Evans RM, Heyman RA, and Mangelsdorf DJ (1995) LXR, a nuclear receptor that defines a distinct retinoid response pathway. *Genes Dev*, 9(9): 1033-1045.
58. Freedman LP and Luisi BF (1993) On the mechanism of DNA binding by nuclear hormone receptors: a structural and functional perspective. *J Cell Biochem*, 51(2): 140-150.
59. Laudet V, Hanni C, Coll J, Catzeflis F, and Stehelin D (1992) Evolution of the nuclear receptor gene superfamily. *EMBO J*, 11(3): 1003-1013.
60. Lieberman BA, Bona BJ, Edwards DP, and Nordeen SK (1993) The constitution of a progesterone response element. *Mol Endocrinol*, 7(4): 515-527.
61. Nelson CC, Hendy SC, Shukin RJ, Cheng H, Bruchovsky N, Koop BF, and Rennie PS (1999) Determinants of DNA sequence specificity of the androgen, progesterone, and glucocorticoid receptors: evidence for differential steroid receptor response elements. *Mol Endocrinol*, 13(12): 2090-2107.
62. Mader S, Kumar V, de Verneuil H, and Chambon P (1989) Three amino acids of the oestrogen receptor are essential to its ability to distinguish an oestrogen from a glucocorticoid-responsive element. *Nature*, 338(6212): 271-274.
63. Fuller PJ (1991) The steroid receptor superfamily: mechanisms of diversity. *FASEB J*, 5(15): 3092-3099.
64. Dahlman-Wright K, Siltala-Roos H, Carlstedt-Duke J, and Gustafsson JA (1990) Protein-protein interactions facilitate DNA binding by the glucocorticoid receptor DNA-binding domain. *J Biol Chem*, 265(23): 14030-14035.
65. Hoglund A and Kohlbacher O (2004) From sequence to structure and back again: approaches for predicting protein-DNA binding. *Proteome Sci*, 2(1): 3.
66. Garner MM and Revzin A (1981) A gel electrophoresis method for quantifying the binding of proteins to specific DNA regions: application to components of the Escherichia coli lactose operon regulatory system. *Nucleic Acids Res*, 9(13): 3047-3060.
67. Latchman DS (1999) The DNA Mobility Shift Assay. *Transcription Factors - A Practical Approach*. Oxford University Press, London.
68. Saluz H and Jost JP (1993) Approaches to Characterize Protein-DNA Interactions *In Vivo*. *Crit Rev Eukaryot Gene Expr*, 3(1): 1-29.
69. Orlando V (2000) Mapping chromosomal proteins in vivo by formaldehyde-crosslinked-chromatin immunoprecipitation. *Trends Biochem Sci*, 25(3): 99-104.
70. Buck MJ and Lieb JD (2004) ChIP-chip: considerations for the design, analysis, and application of genome-wide chromatin immunoprecipitation experiments. *Genomics*, 83(3): 349-360.

71. Luisi BF, Xu WX, Otwinowski Z, Freedman LP, Yamamoto KR, and Sigler PB (1991) Crystallographic analysis of the interaction of the glucocorticoid receptor with DNA. *Nature*, 352(6335): 497-505.
72. Varani G, Chen Y, and Lepper TC (2004) NMR studies of protein-nucleic acid interactions. *Methods Mol Biol*, 278: 289-312.
73. Carey M and Smale ST (2000) Identification and analysis of distant control regions. *Transcriptional Regulation in Eukariotes. Concepts, Strategies and Techniques*. Cold Spring Harbor Laboratory Press, New York.
74. Godchaux W, Adamson SO, and Herbert E (1967) Effects of cycloheximide on polyribosome function in reticulocytes. *J Mol Biol*, 27(1): 57-72.
75. Kufe DW, Pollock RE, Weichselbaum RR, Bast C Jr, Gansler TS, Holland James and Frei III E (2003) Cancer Biology. *Cancer Medicine*. BC Decker Inc, Hamilton (Canada).
76. Pabo CO and Sauer RT (1992) Transcription factors: structural families and principles of DNA recognition. *Annu Rev Biochem*, 61: 1053-1095.
77. Tompa M, Li N, Tompa M, Li N, Bailey TL, Church GM, De Moor B, Eskin E et al. (2005) Assessing computational tools for the discovery of transcription factor binding sites. *Nat Biotechnol*, 23(1): 137-144.
78. Day WH and McMorris FR (1992) Critical comparison of consensus methods for molecular sequences. *Nucleic Acids Res*, 20(5): 1093-1099.
79. Prestidge DS (1991) SIGNAL SCAN: a computer program that scans DNA sequences for eukaryotic transcriptional elements. *Comput Appl Biosci*, 7(2): 203-206.
80. Quandt K, Frech K, Karas H, Wingender E, and Werner T (1995) MatInd and MatInspector: new fast and versatile tools for detection of consensus matches in nucleotide sequence data. *Nucleic Acids Res*, 23(23): 4878-4884.
81. Bulyk ML (2003) Computational prediction of transcription-factor binding site locations. *Genome Biol*, 5(1): 201.
82. Matys V, Fricke E, Geffers R, Gossling E, Haubrock M, Hehl R, Hornischer K, Karas D et al. (2003) TRANSFAC: transcriptional regulation, from patterns to profiles. *Nucleic Acids Res*, 31(1): 374-378.
83. Sandelin A, Alkema W, Engstrom P, Wasserman WW, and Lenhard B (2004) JASPAR: an open-access database for eukaryotic transcription factor binding profiles. *Nucleic Acids Res*, 32(Database issue): D91-D94.
84. Cartharius K, Frech K, Grote K, Klocke B, Haltmeier M, Klingenhoff A, Frisch M, Bayerlein M et al. (2005) MatInspector and beyond: promoter analysis based on transcription factor binding sites. *Bioinformatics*, 21(13): 2933-2942.

85. Kel AE, Gossling E, Reuter I, Cheremushkin E, Kel-Margoulis OV, and Wingender E (2003) MATCH: A tool for searching transcription factor binding sites in DNA sequences. *Nucleic Acids Res*, 31(13): 3576-3579.
86. Smith AD, Sumazin P, Xuan Z, and Zhang MQ (2006) DNA motifs in human and mouse proximal promoters predict tissue-specific expression. *Proc Natl Acad Sci USA*, 103(16): 6275-6280.
87. Schneider TD and Stephens RM (1990) Sequence logos: a new way to display consensus sequences. *Nucleic Acids Res*, 18(20): 6097-6100.
88. Cornish-Bowden A (1985) Nomenclature for incompletely specified bases in nucleic acid sequences: recommendations 1984. *Nucleic Acids Res*, 13(9): 3021-3030.
89. Markov A (1971) Extension of the limit theorems of probability theory to a sum of variables connected in a chain. *Dynamic Probabilistic Systems, volume 1: Markov Chains*. John Wiley and Sons.
90. Rajapakse JC and Loi Sy Ho (2005) Markov Encoding for Detecting Signals in Genomic Sequences. *IEEE/ACM Trans Comput Biol Bioinform*, 2(2): 131-142.
91. Eddy SR (2005) Hidden Markov models. *Curr Opin Struct Biol*, 6(3): 361-365.
92. Shida K (2006) GibbsST: a Gibbs sampling method for motif discovery with enhanced resistance to local optima. *BMC Bioinformatics*, 7: 486.
93. Thompson W, Rouchka EC, and Lawrence CE (2003) Gibbs Recursive Sampler: finding transcription factor binding sites. *Nucleic Acids Res*, 31(13): 3580-3585.
94. Thijs G, Marchal K, Lescot M, Rombauts S, De Moor B, Rouze P, and Moreau Y (2002) A Gibbs sampling method to detect overrepresented motifs in the upstream regions of coexpressed genes. *J Comput Biol*, 9(2): 447-464.
95. Murakami K, Kojima T, and Sakaki Y (2004) Assessment of clusters of transcription factor binding sites in relationship to human promoter, CpG islands and gene expression. *BMC Genomics*, 5(1): 16.
96. Pudimat R, Schukat-Talamazzini EG, and Backofen R (2005) A multiple-feature framework for modelling and predicting transcription factor binding sites. *Bioinformatics*, 21(14): 3082-3088.
97. Aeling KA, Steffen NR, Johnson M, Hatfield GW, Lathrop RH, and Senear DF (2007) DNA deformation energy as an indirect recognition mechanism in protein-DNA interactions. *IEEE/ACM Trans Comput Biol Bioinform*, 4(1): 117-125.
98. Li X, Zhong S, and Wong WH (2005) Reliable prediction of transcription factor binding sites by phylogenetic verification. *Proc Natl Acad Sci U S A*, 102(47): 16945-16950.

-
99. Kim SY and Kim Y (2006) Genome-wide prediction of transcriptional regulatory elements of human promoters using gene expression and promoter analysis data. *BMC Bioinformatics*, 7: 330.
 100. Grau J, Ben-Gal I, Posch S, and Grosse I (2006) VOMBAT: prediction of transcription factor binding sites using variable order Bayesian trees. *Nucleic Acids Res*, 34(W): 529-533.
 101. Reddy TE, DeLisi C, and Shakhnovich BE (2007) Binding site graphs: a new graph theoretical framework for prediction of transcription factor binding sites. *PLoS Comput Biol*, 3(5): e90.
 102. Jolly ER, Chin CS, Herskowitz I, and Li H (2005) Genome-wide identification of the regulatory targets of a transcription factor using biochemical characterization and computational genomic analysis. *BMC Bioinformatics*, 6: 275.
 103. Vega VB, Lin CY, Lai KS, Kong SL, Xie M, Su X, Teh HF, Thomsen JS *et al.* (2006) Multiplatform genome-wide identification and modeling of functional human estrogen receptor binding sites. *Genome Biol*, 7(9): R82.
 104. Tang S, Tan SL, Ramadoss SK, Kumar AP, Tang MH, and Bajic VB (2004) Computational method for discovery of estrogen responsive genes. *Nucleic Acids Res*, 32(21): 6212-6217.
 105. Podvinec M, Kaufmann MR, Handschin C, and Meyer UA (2002) NUBIScan, an in silico approach for prediction of nuclear receptor response elements. *Mol Endocrinol*, 16(6): 1269-1279.
 106. Sandelin A and Wasserman WW (2005) Prediction of nuclear hormone receptor response elements. *Mol Endocrinol*, 19(3): 595-606.
 107. Favorov AV, Gelfand MS, Gerasimova AV, Ravcheev DA, Mironov AA, and Makeev VJ (2005) A Gibbs sampler for identification of symmetrically structured, spaced DNA motifs with improved estimation of the signal length. *Bioinformatics*, 21(10): 2240-2245.
 108. Bi C, Leeder JS, and Vyhlidal CA (2008) A comparative study on computational two-block motif detection: algorithms and applications. *Mol Pharm*, 5(1): 3-16.
 109. Mitchell T (1997) Machine Learning. McGraw-Hill Science/Engineering/Math.
 110. MacKay D (2002) Information Theory, Inference, and Learning Algorithms. Cambridge University Press.
 111. Ye J, McGinnis S, and Madden TL (2006) BLAST: improvements for better sequence analysis. *Nucleic Acids Res*, 34(W): 6-9.
 112. Deschamps JP, Bioul G and Sutter GD (2006) Synthesis of arithmetic circuits: FPGA, ASIC, and embedded systems. Wiley-Interscience.
 113. Cofer RC and Harding BF (2005) Rapid system prototyping with FPGAs. Newnes.

114. Wolf W (2004) FPGA-Based System Design. Prentice Hall PTR, Upper Saddle River, NJ.
115. Ciletti MD (2003) Advanced digital system design with Verilog HDL. Prentice Hall.
116. Sotiriades E and Dollas A (2007) A General Reconfigurable Architecture for the BLAST Algorithm. *J VLSI Sig Process S*, 48(3): 189-208.
117. Stamatakis A, Blagojevic F, Nikolopoulos DS, and Antonopoulos CD (2007) Exploring New Search Algorithms and Hardware for Phylogenetics: RAxML Meets the IBM Cell. *J VLSI Sig Process S*, 48(3): 271-286.
118. Gu Y, Van Court T, DiSabello D, and Herbordt MC (2005) Preliminary report: FPGA acceleration of molecular dynamics computations. *In Proc of 13th Annual IEEE Symposium on Field-Programmable Custom Computing Machines*, 269-270.
119. Beato M (1989) Gene regulation by steroid hormones. *Cell*, 56(3): 335-344.
120. Geserick C, Meyer HA, and Haendler B (2005) The role of DNA response elements as allosteric modulators of steroid receptor function. *Mol Cell Endocrinol*, 236(1-2): 1-7.
121. Smirnov AN (2002) Nuclear receptors: nomenclature, ligands, mechanisms of their effects on gene expression. *Biochemistry (Mosc)*, 67(9): 957-977.
122. Barbulescu K, Geserick C, Schuttke I, Schleuning WD, and Haendler B (2001) New androgen response elements in the murine pem promoter mediate selective transactivation. *Mol Endocrinol*, 15(10): 1803-1816.
123. Truss M, Chalepakakis G, and Beato M (1990) Contacts between steroid hormone receptors and thymines in DNA: an interference method. *Proc Natl Acad Sci U S A*, 87(18): 7180-7184.
124. Thackray VG, Lieberman BA, and Nordeen SK (1998) Differential gene induction by glucocorticoid and progesterone receptors. *J Steroid Biochem Mol Biol*, 66(4): 171-178.
125. Roche PJ, Hoare SA, and Parker MG (1992) A consensus DNA-binding site for the androgen receptor. *Mol Endocrinol*, 6(12): 2229-2235.
126. Haelens A, Verrijdt G, Callewaert L, Christiaens V, Schauwaers K, Peeters B, Rombauts W, and Claessens F (2003) DNA recognition by the androgen receptor: evidence for an alternative DNA-dependent dimerization, and an active role of sequences flanking the response element on transactivation. *Biochem J*, 369(1): 141-151.
127. Zhou Z, Corden JL, and Brown TR (1997) Identification and characterization of a novel androgen response element composed of a direct repeat. *J Biol Chem*, 272(13): 8227-8235.

128. GuhaThakurta D (2006) Computational identification of transcriptional regulatory elements in DNA sequence. *Nucleic Acids Res*, 34(12): 3585-3598.
129. Ferretti V, Poitras C, Bergeron D, Coulombe B, Robert F, and Blanchette M (2007) PReMod: a database of genome-wide mammalian cis-regulatory module predictions. *Nucleic Acids Res*, 35(D): 122-126.
130. Stepanova M, Lin F, and Lin V (2006) Establishing a Statistic Model for Recognition of Steroid Hormone Response Elements. *Comput Biol Chem*, 30(5): 339-347.
131. Gusfield D (1997) Algorithms on Strings, Trees, and Sequences. Cambridge University Press.
132. Rice J (1994) Mathematical Statistics and Data Analysis. Duxbury Press.
133. Salkind NJ (2006) Encyclopedia of Measurement and Statistics. Sage Publications.
134. Benjamini Y and Hochberg Y (1995) Controlling the false discovery rate: a practical and powerful approach to multiple testing. *J Roy Statist Soc Ser B*, 57(1): 289-300.
135. Bird AP (1986) CpG-rich islands and the function of DNA methylation. *Nature*, 321(6067): 209-213.
136. Stepanova M, Tyazhelova T, Skoblov M, and Baranova A (2005) A comparative analysis of relative occurrence of transcription factor binding sites in vertebrate genomes and gene promoter areas. *Bioinformatics*, 21(9): 1789-1796.
137. Eddy SR (1998) Profile hidden Markov models. *Bioinformatics*, 14(9): 755-763.
138. Clote P (2000) Computational molecular biology : an introduction. John Wiley, New York.
139. Hagan M, Demuth H and Beale M (1996) Neural network design. PWS Pub, Boston.
140. Haykin S (1999) Neural Networks: A Comprehensive Foundation. Prentice Hall, New Jersey.
141. Pinkus A (1999) Approximation theory of the MLP model in neural networks. *Acta Numerica*, 8: 143-196.
142. Cybenko G (1989) Approximation by superposition of a sigmoidal function. *Math Control Signal Systems*, 2(4): 303-314.
143. Hard T, Kellenbach E, Boelens R, Maler BA, Dahlman K, Freedman LP, Carlstedt-Duke J, Yamamoto KR et al. (1990) Solution structure of the glucocorticoid receptor DNA-binding domain. *Science*, 249(4965): 157-160.

144. Schwabe JW, Neuhaus D, and Rhodes D (1990) Solution structure of the DNA-binding domain of the oestrogen receptor. *Nature*, 348(6300): 458-461.
145. Alroy I and Freedman LP (1992) DNA binding analysis of glucocorticoid receptor specificity mutants. *Nucleic Acids Res*, 20(5): 1045-1052.
146. Aranda A and Pascual A (2001) Nuclear hormone receptors and gene expression. *Physiol Rev*, 81(3): 1269-1304.
147. Pollack JB (1991) The introduction of dynamical recognizers. *Machine Learning*, 7: 227-252.
148. Simpson JJ and McIntire TJ (2001) A Recurrent Neural Network Classifier for Improved Retrievals of Areal Extent of Snow Cover. *IEEE Trans Geosci Remote Sens*, 39(10): 2135-2147.
149. Guler I and Ibeyli ED (2006) A recurrent neural network classifier for Doppler ultrasound blood flow signals. *Pattern Recognition Letters*, 27(13): 1560-1571.
150. Hammer B (2000) Learning With Recurrent Neural Networks. London.
151. Elman J (1990) Finding structure in time. *Cognitive Sci*, 14: 179-211.
152. Hopfield JJ (1982) Neural networks and physical systems with emergent collective computational abilities. *Proc Natl Acad Sci U S A*, 79(8): 2554-2558.
153. Li JH, Michel A, and Porod W (1989) Analysis and synthesis of a class of neural networks: linear systems operating on a closed hypercube. *IEEE Trans Circuits Syst*, 36(11): 1405-1422.
154. Gronemeyer H (1992) Control of transcription activation by steroid hormone receptors. *FASEB J*, 6(8): 2524-2529.
155. Thackray VG, Lieberman BA, and Nordeen SK (1998) Differential gene induction by glucocorticoid and progesterone receptors. *J Steroid Biochem Mol Biol*, 66(4): 171-178.
156. Stepanova M, Lin F, and Lin V (2007) A Hopfield Neural Classifier and Its FPGA Implementation for Identification of Symmetrically Structured DNA Motifs. *Journal of VLSI Signal Processing Systems*, 48(3): 239-254.
157. Needleman SB and Wunsch CD (1970) A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3): 443-453.
158. Reid KJ, Hendy SC, Saito J, Sorensen P, and Nelson CC (2001) Two classes of androgen receptor elements mediate cooperativity through allosteric interactions. *J Biol Chem*, 276(4): 2943-2952.
159. Claessens F, Verrijdt G, Schoenmakers E, Haelens A, Peeters B, Verhoeven G, and Rombauts W (2001) Selective DNA binding by the androgen receptor as a

- mechanism for hormone-specific gene regulation. *J Steroid Biochem Mol Biol*, 76(1-5): 23-30.
160. Stepanova M, Lin F, and Lin V (2007) A Two-phase ANN Method for Genome-wide Detection of Hormone Response Elements. *Lecture Notes in Bioinformatics*, 4774: 19-29.
 161. Britannica (2004) Encyclopedia Britannica, Inc.
 162. Workman CT and Stormo GD (2000) ANN-Spec: a method for discovering transcription factor binding sites with improved specificity. *Pac Symp Biocomput*, 467-478.
 163. Shigeo A (2005) Support vector machines for pattern classification. Springer.
 164. Jantzen K, Fritton HP, Igo-Kemenes T, Espel E, Janich S, Cato AC, Mugele K, and Beato M (1987) Partial overlapping of binding sequences for steroid hormone receptors and DNaseI hypersensitive sites in the rabbit uteroglobin gene region. *Nucleic Acids Res*, 15(11): 4535-4552.
 165. Claessens F, Alen P, Devos A, Peeters B, Verhoeven G, and Rombauts W (1996) The androgen-specific probasin response element 2 interacts differentially with androgen and glucocorticoid receptors. *J Biol Chem*, 271(32): 19013-19016.
 166. von der Ahe D, Renoir JM, Buchou T, Baulieu EE, and Beato M (1986) Receptors for glucocorticosteroid and progesterone recognize distinct features of a DNA regulatory element. *Proc Natl Acad Sci U S A*, 83(9): 2817-2821.
 167. Lamian V, Gonzalez BY, Michel FJ, and Simmen RC (1993) Non-consensus progesterone response elements mediate the progesterone-regulated endometrial expression of the uteroferrin gene. *J Steroid Biochem Mol Biol*, 46(4): 439-450.
 168. Drouin J, Trifiro MA, Plante RK, Nemer M, Eriksson P, and Wrange Oe (1989) Glucocorticoid receptor binding to a specific DNA sequence is required for hormone-dependent repression of pro-opiomelanocortin gene transcription. *Mol Cell Biol*, 9(12): 5305-5314.
 169. Ma T, Copland JA, Brasier AR, and Thompson EA (2000) A novel glucocorticoid receptor binding element within the murine c-myc promoter. *Mol Endocrinol*, 14(9): 1377-1386.
 170. Moens U, Subramaniam N, Johansen B, Johansen T, and Traavik T (1994) A steroid hormone response unit in the late leader of the noncoding control region of the human polyomavirus BK confers enhanced host cell permissivity. *J Virol*, 68(4): 2398-2408.
 171. Welte T, Philipp S, Cairns C, Gustafsson JA, and Doppler W (1993) Glucocorticoid receptor binding sites in the promoter region of milk protein genes. *J Steroid Biochem Mol Biol*, 47(1-6): 75-81.

172. Kolla V, Robertson NM, and Litwack G (1999) Identification of a mineralocorticoid/glucocorticoid response element in the human Na/K ATPase α 1 gene promoter. *Biochem Biophys Res Commun*, 266(1): 5-14.
173. Verrijdt G, Schauwaers K, Haelens A, Rombauts W, and Claessens F (2002) Functional interplay between two response elements with distinct binding characteristics dictates androgen specificity of the mouse sex-limited protein enhancer. *J Biol Chem*, 277(38): 35191-35201.
174. Schoneveld O, Gaemers I, and Lamers W (2004) Mechanisms of glucocorticoid signalling. *Biochim Biophys Acta*, 1680(2): 114-128.
175. Yan W, Imanishi M, Futaki S, and Sugiura Y (2007) Alpha-helical linker of an artificial 6-zinc finger peptide contributes to selective DNA binding to a discontinuous recognition sequence. *Biochemistry*, 46(29): 8517-8524.
176. Majumder P, Pradhan SK, Devi PG, Pal S, and Dasgupta D (2007) Chromatin as a target for the DNA-binding anticancer drugs. *Subcell Biochem*, 41: 145-189.
177. Robins DM, Scheller A, and Adler AJ (1994) Specific steroid response from a nonspecific DNA element. *J Steroid Biochem Mol Biol*, 49(4-6): 251-255.
178. Robertson TA and Varani G (2007) An all-atom, distance-dependent scoring function for the prediction of protein-DNA interactions from structure. *Proteins*, 66(2): 359-374.
179. Luscombe NM, Laskowski RA, and Thornton JM (2001) Amino acid-base interactions: a three-dimensional analysis of protein-DNA interactions at an atomic level. *Nucleic Acids Res*, 29(13): 2860-2874.
180. The UniProt Consortium (2007) The Universal Protein Resource (UniProt). *Nucleic Acids Res*, 35(D): 193-197.
181. Luisi BF, Xu WX, Otwinowski Z, Freedman LP, Yamamoto KR, and Sigler PB (1991) Crystallographic analysis of the interaction of the glucocorticoid receptor with DNA. *Nature*, 352(6335): 497-505.
182. Ormondi A and Rajapakse J (2006) FPGA implementations of neural networks. Springer, Netherlands.
183. Hammerstrom D (1991) A highly parallel digital architecture for neural network simulation. *VLSI for Artificial Intelligence and Neural Networks*. Plenum press.
184. Weinstein RK and Lee RH (2006) Architectures for high-performance FPGA implementations of neural models. *J Neural Eng*, 3: 21-34.
185. Upegui A, Pena-Reyes CA, and Sanchez E (2005) An FPGA platform for on-line topology exploration of spiking neural networks. *Microprocessors and Microsystems*, 29(5): 211-223.
186. Clare PJ, Gulley JW, Hickman D, and Smith MI (1997) Design and tuning of FPGA implementations of neural networks. *Proc. SPIE*, 3069: 129-136.

187. Maeda Y and Tada T (2003) FPGA implementation of a pulse density neural network with learning ability using simultaneous perturbation. *IEEE Trans Neural Netw*, 14(3): 688-695.
188. Cauwenberghs G (1996) An analog VLSI recurrent neural network learning a continuous-time trajectory. *IEEE Trans Neural Netw*, 7(2): 346-361.
189. Zhu J and Sutton P (2003) FPGA Implementation of Neural Networks - A Survey of a Decade of Progress. In *Proc of 13th International Conference on Field-Programmable Logic and Applications (FPL 2003)*, 1062-1066.
190. Ahola TM, Purmonen S, Pennanen P, Zhuang YH, Tuohimaa P, and Ylikomi T (2002) Progesterin upregulates G-protein-coupled receptor 30 in breast cancer cells. *Eur J Biochem*, 269(10): 2485-2490.
191. Vassen L, Wegrzyn W, and Klein-Hitpass L (1999) Human insulin receptor substrate-2 (IRS-2) is a primary progesterone response gene. *Mol Endocrinol*, 13(3): 485-494.
192. Tsuchiya S, Tanaka S, Sugimoto Y, Katsuyama M, Ikegami R, and Ichikawa A (2003) Identification and characterization of a novel progesterone receptor-binding element in the mouse prostaglandin E receptor subtype EP2 gene. *Genes Cells*, 8(9): 747-758.
193. Kester HA, van der Leede BM, van der Saag PT, and van der Burg B (1997) Novel progesterone target genes identified by an improved differential display technique suggest that progestin-induced growth inhibition of breast cancer cells coincides with enhancement of differentiation. *J Biol Chem*, 272(26): 16637-16643.
194. Cheon YP, Xu X, Bagchi MK, and Bagchi IC (2003) Immune-responsive gene 1 is a novel target of progesterone receptor and plays a critical role during implantation in the mouse. *Endocrinology*, 144(12): 5623-5630.
195. Purmonen S, Ahola TM, Pennanen P, Aksenov N, Zhuang YH, Tuohimaa P, and Ylikomi T (2002) HDLG5/KIAA0583, encoding a MAGUK-family protein, is a primary progesterone target gene in breast cancer cells. *Int J Cancer*, 102(1): 1-6.
196. Tang M, Mazella J, Gao J, and Tseng L (2002) Progesterone receptor activates its promoter activity in human endometrial stromal cells. *Mol Cell Endocrinol*, 192(1-2): 45-53.
197. Leo JC, Wang SM, Guo CH, Aw SE, Zhao Y, Li JM, Hui KM, and Lin VC (2005) Gene regulation profile reveals consistent anticancer properties of progesterone in hormone-independent breast cancer cells transfected with progesterone receptor. *Int J Cancer*, 117(4): 561-568.
198. Wheeler DL, Barrett T, Benson DA, Bryant SH, Canese K, Chetvernin V, et al. (2006) Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res*, 34(D): 173-180.

199. Tsai SY, Tsai MJ, and O'Malley BW (1989) Cooperative binding of steroid hormone receptors contributes to transcriptional synergism at target enhancer elements. *Cell*, 57(3): 443-448.
200. Chenna R, Sugawara H, Koike T, Lopez R, Gibson TJ, Higgins DG, and Thompson JD (2003) Multiple sequence alignment with the Clustal series of programs. *Nucleic Acids Res*, 31(13): 3497-3500.
201. Lapp CA, Lohse JE, Lewis JB, Dickinson DP, Billman M, Hanes PJ, and Lapp DF (2003) The effects of progesterone on matrix metalloproteinases in cultured human gingival fibroblasts. *J Periodontol*, 74(3): 277-288.
202. Bjorling DE, Beckman M, Clayton MK, and Wang ZY (2002) Modulation of nerve growth factor in peripheral organs by estrogen and progesterone. *Neuroscience*, 110(1): 155-167.
203. Chimpanzee Sequencing and Analysis Consortium (2005) Initial sequence of the chimpanzee genome and comparison with the human genome. *Nature*, 437(7055): 69-87.
204. Milde-Langosch K (2005) The Fos family of transcription factors and their role in tumourigenesis. *Eur J Cancer*, 41(16): 2449-2461.
205. Hoffmann E, Thiefes A, Buhrow D, Dittrich-Breiholz O, Schneider H, Resch K, and Kracht M (2005) MEK1-dependent delayed expression of Fos-related antigen-1 counteracts c-Fos and p65 NF-kappaB-mediated interleukin-8 transcription in response to cytokines or growth factors. *J Biol Chem*, 280(10): 9706-9718.
206. Levy S, Sutton G, Ng PC, Feuk L, Halpern AL, Walenz BP, et al. (2007) The Diploid Genome Sequence of an Individual Human. *PLoS Biol*, 5(10): e254.
207. Bono HU (2005) SayaMatcher: Genome scale organization and systematic analysis of nuclear receptor response elements. *Gene*, 364 :74-78.
208. Tirosch I, Berman J, and Barkai N (2007) The pattern and evolution of yeast promoter bendability. *Trends Genet*, 23(7): 318-321.
209. Horie-Inoue K, Takayama K, Bono HU, Ouchi Y, Okazaki Y, and Inoue S (2006) Identification of novel steroid target genes through the combination of bioinformatics and functional analysis of hormone response elements. *Biochem Biophys Res Commun*, 339(1): 99-106.
210. Leo JC and Lin VC (2008) The activities of progesterone receptor isoform A and B are differentially modulated by their ligands in a gene-selective manner. *Int J Cancer*, 122(1): 230-243.
211. Ureta-Vidal A, Ettwiller L, and Birney E (2003) Comparative genomics: genome-wide analysis in metazoan eukaryotes. *Nat Rev Genet*, 4(4): 251-262.
212. Hardison RC (2000) Conserved noncoding sequences are reliable guides to regulatory elements. *Trends Genet*, 16(9): 369-372.

213. Lenhard B, Sandelin A, Mendoza L, Engstrom P, Jareborg N, and Wasserman WW (2003) Identification of conserved regulatory elements by comparative genome analysis. *J Biol*, 2(2): 13.
214. Brudno M, Do CB, Cooper GM, Kim MF, Davydov E, Green ED, Sidow A, Batzoglou S et al. (2003) LAGAN and Multi-LAGAN: efficient tools for large-scale multiple alignment of genomic DNA. *Genome Res*, 13(4): 721-731.
215. Montgomery SB, Astakhova T, Bilenky M, Birney E, Fu T, Hassel M, Melsopp C, Rak M et al. (2004) Sockeye: a 3D environment for comparative genomics. *Genome Res*, 14(5): 956-962.
216. Loots GG, Ovcharenko I, Pachter L, Dubchak I, and Rubin EM (2002) rVista for comparative sequence-based discovery of functional transcription factor binding sites. *Genome Res*, 12(5): 832-839.
217. Mustonen V and Lassig M (2005) Evolutionary population genetics of promoters: predicting binding sites and functional phylogenies. *Proc Natl Acad Sci U S A*, 102(44): 15936-15941.
218. Elnitski L, Hardison RC, Li J, Yang S, Kolbe D, Eswara P, O'Connor MJ, Schwartz S et al. (2003) Distinguishing regulatory DNA from neutral sites. *Genome Res*, 13(1): 64-72.
219. Kel-Margoulis OV, Kel AE, Reuter I, Deineko IV, and Wingender E (2002) TRANSCompel: a database on composite regulatory elements in eukaryotic genes. *Nucleic Acids Res*, 30(1): 332-334.
220. Rajewsky N, Vergassola M, Gaul U, and Siggia ED (2002) Computational detection of genomic cis-regulatory modules applied to body patterning in the early *Drosophila* embryo. *BMC Bioinformatics*, 3(1): 30.
221. Carvalho AM, Freitas AT, Oliveira AL, and Sagot MF (2006) An efficient algorithm for the identification of structured motifs in DNA promoter sequences. *IEEE/ACM Trans Comput Biol Bioinform*, 3(2): 126-140.
222. Gao X, Loggie BW, and Nawaz Z (2002) The roles of sex steroid receptor coregulators in cancer. *Mol Cancer*, 1: 7.
223. Beato M and Klug J (2000) Steroid hormone receptors: an update. *Hum Reprod Update*, 6(3): 225-236.
224. Furringer M, Fuchs G, Steininger A and Kempf G (2006) VLSI Implementation of a Fault-Tolerant Distributed Clock Generation. In *Proc of 21st IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DFT'06)*, 563-571.
225. API NetWorks Accelerates Use of HyperTransport™ Technology with Launch of Industry's First HyperTransport Technology-to-PCI Bridge Chip. Press-Release of the Hyper-Transport Consortium. Concord MA, USA, 2001.