FPGA-BASED ACCELERATOR FOR THE GENERATION OF PSEUDO-AMINO ACID COMPOSITION

By

CHING CHEE CHOW

A Dissertation submitted for partial fulfilment of the requirement for the degree of Master of Science (Microelectronic Engineering)

August 2015

ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my research supervisor, Dr. Bakhtiar Affendi Rosdi, for his dedication and commitment in providing guidance and support throughout the research. His willingness to impart knowledge and share his experience played a major role in ensuring the success of the research. He had always offered constructive feedback and insightful opinion whilst affording me the space to be creative in my work. I had learned a lot from him in both theoretical and practical aspects of the research. I would also like to thank the School of Electrical and Electronic Engineering of Universiti Sains Malaysia, and Dr. Bakhtiar in particular, for assisting in acquiring the hardware necessary for the research. The hardware plays an important role in achieving the objectives of the research. I am grateful to the Ministry of Education of Malaysia for offering financial support in the form of the MyMaster scholarship. The financial aid had given me peace of mind and motivation to work hard and complete my study successfully. Last but not least, I am also grateful to my family and friends for being a pillar of strength to me during the entire course of the research. Their encouragement and support had given me the inspiration to perform to the best of my ability.

TABLE OF CONTENTS

CHAPTER 3 - METHODOLOGY

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AA	Amino Acid
AAC	Amino Acid Composition
AAIndex1	Amino Acid Index One
ASM	Algorithmic State Machine
BLAST	Basic Local Alignment Search Tool
CAST	Complexity Analysis of Sequence Tracts
CD	Covariance Discriminant
CDM	Complexity-based Distance Measure
CPU	Central Processor Unit
CSV	Comma-separated Value
DIP	Dual-in line Package
DNA	Deoxyribonucleic Acid
DSP	Digital Signal Processor
FASTA	Fast-All
FPGA	Field Programmable Gate Array
GlimmerHMM	Gene Locator and Interpolated Markov Model ER Hidden Markov Model
GUI	Graphic User Interface
HDL	Hardware Description Language
I/O	Input/Output
IC	Integrated Circuit
IMM	Interpolated Markov Model
JTAG	Joint Test Action Group
K-NN	K-Nearest Neighbor
LE	Logic Element
LED	Light-emitting Diode
LSB	Least Significant Bit

LUT	Look-up Table
LZ	Lempel-Ziv
MAFFT	Multiple Alignment using Fast Fourier Transform
MB	Megabyte
MIF	Memory Initialization File
mRMR	Minimum Redundancy Maximum Relevance
MS	Mass Spectrometry
MSA	Multiple Sequence Alignment
MSB	Most Significant Bit
NN-CDM	Nearest Neighbor-Complexity-based Distance Measure
PseAAC	Pseudo-Amino Acid Composition
PLL	Phase-locked Loop
PSI	Position-specific Iterated
RAM	Random Access Memory
RNA	Ribonucleic Acid
RTL	Register Transfer Level
sed	Stream editor
SOF	SRAM Object File
SRAM	Static Random Access Memory
SVM	Support Vector Machine
UI	User Interface

PEMECUT BERASASKAN FPGA UNTUK PENJANAAN KOMPOSISI ASID AMINO PSEUDO

ABSTRAK

Pengurangan jurang antara bilangan protein baru yang belum dicirikan dan yang telah dikenali di dalam bank data protein telah muncul sebagai salah satu cabaran terbesar era pasca genomik. Permintaan kian meningkat untuk teknik-teknik yang dapat meramal ciri-ciri protein dengan cekap dan tepat berdasarkan maklumat urutan protein sahaja. Komposisi Asid Amino Pseudo (PseAAC) telah muncul sebagai teknik pemodelan yang berupaya menggabungkan maklumat urutan protein terpilih dalam model diskret. PseAAC telah digunakan secara meluas dalam ujikaji protein melalui pelbagai perisian penjana PseAAC. Oleh sebab penjanaan PseAAC lazimnya melibatkan pemprosesan data berskala besar, tempoh pemprosesan amat penting. Prospek untuk mengurangkan tempoh tersebut terhad kerana proses perisian lazimnya berjujukan. Maka, perkakasan yang boleh diaturcara seperti *Field Programmable Gate Array* (FPGA) muncul sebagai alternatif baru dengan keupayaan pemprosesan selari yang dapat mempercepat penghitungan PseAAC. Dalam penyelidikan ini, suatu pemecut berasaskan FPGA untuk penjanaan PseAAC telah diperkenalkan. Penjana tersebut terdiri daripada beberapa modul. Untuk mempercepat proses, dua modul yang paling intensif dalam penghitungan, iaitu *Sum-of-Small-T* dan *T-u-minus-20*, direka untuk pelaksanaan secara selari. Penjana tersebut direalisasikan melalui FPGA Altera Cyclone III. Proses berjaya dipercepat sehingga 31.5 kali ganda berbanding suatu penjana PseAAC berasaskan perisian Perl. Kesimpulannya, pengurangan tempoh penghitungan yang ketara telah dicapai melalui rekabentuk penjana PseAAC yang menggunakan kebolehan pemprosesan selari FPGA.

FPGA-BASED ACCELERATOR FOR THE GENERATION OF PSEUDO-AMINO ACID COMPOSITION

ABSTRACT

One of the biggest challenges in protein prediction post genomic age is narrowing the gap between the number of newly discovered and uncharacterized proteins and the number of known proteins in protein data banks. This leads to increased demand for efficient techniques to accurately predict protein attributes based solely on its sequence-order information. The Pseudo-Amino Acid Composition (PseAAC) is a modeling technique that incorporates, selectively, sequence-order information of a protein into a discrete model. PseAAC has been applied in numerous protein-related researches using various software-based PseAAC generators. Since this often involves large-scale data processing, computation time is of the essence. The prospect of further reducing computation time of the software is limited due to the sequential nature of software execution. Alternative platform such as programmable hardware has emerged as a solution to this bottleneck. Programmable hardware such as Field Programmable Gate Array (FPGA) enables parallel processing that speeds up computation of PseAAC. In this research, an FPGA-based PseAAC generator architecture is proposed. The architecture consists of several modules. To speed up computation, the two most computation-heavy modules of the architecture, the Sum-of-Small-T and T-u-minus-20, are designed to run in parallel. The generator is realized on the Altera Cyclone III FPGA and achieves computation speed increase of up to 31.5 times over a Perl-based PseAAC generator. In conclusion, significant computation speed improvement is achieved by designing the PseAAC generator to capitalize on the parallel processing capability of the FPGA.

CHAPTER 1

INTRODUCTION

1.1 Background

Proteins are large molecules made up of one or more chains of amino acids bonded together by peptide linkages. It contains elements such as carbon, hydrogen, nitrogen, oxygen and sulfur. A protein chain consists of amino acid residues that can be categorized into 20 naturally-occurring, or native, amino acid types (Mondal & Pai, 2014). The sequence arrangement of these residues plays an important role in determining the structural and functional attributes of a protein.

Protein-based genomics, or proteomics, is the study of proteins, particularly on their structures and functions. Proteomics enables scientists to obtain insights into the interactive relationship between protein and cell. This information is important especially in drug discovery (Schirle, et al., 2012) and cancer studies (Cho, 2014). Proteomics has emerged as an important field, especially in biology and medicine. Research in cell behavior based on genes alone is no longer sufficient and needs to be expanded to include proteins given their close correlation to cell activities.

The increased importance of proteomics and the advancement in bioinformatics applications are inter-dependable. Bioinformatics is a multi-disciplinary field that applies computing, statistics, mathematics and engineering techniques in biological data processing. As proteomics often involves large amount of data, it is highly desirable to handle this data in a computational and mathematical manner in order to increase

processing efficiency and speed. The ability to interpret large-scale data, also known as Mass Spectrometry (MS), is one of the determining factor in the growth of proteomics. The rapid expansion of MS data in proteomics research has stimulated the growth of bioinformatics applications (Colangelo, et al., 2015).

Systems biology is the study of the relationship among components of a biological system. A system may consists of components that range from a small number of protein molecules to groups of cells that perform specific functions in the system (Weston & Hood, 2004). Proteomics plays an important role in systems biology by providing MS-based analytical methods to identify components in a biological system (Sabido, et al., 2012). With improved understanding of systems biology, scientists and doctors are able to find solutions for predicting, preventing and remedying health issues.

The importance of proteomics, bioinformatics and systems biology and their inter-dependency has been discussed. In the post-genomic age, the number of new and uncharacterized proteins being discovered is increasing rapidly (Liu, et al., 2015). By decoding their structural and functional attributes, the new proteins may provide solutions to the discovery of new drugs. The most conventional method in extracting such information is by conducting biochemical experiments which are often expensive and time-consuming. As such, various alternatives have been proposed to predict attributes of new proteins in an efficient and timely manner. One of the most prominent alternatives is the use of Pseudo-Amino Acid Composition (PseAAC) in protein prediction (Mandal, et al., 2015). PseAAC is widely adopted for its modeling simplicity and ability to retain some sequence-order information essential to predicting protein attributes. Due to wide application of PseAAC, numerous software-based PseAAC generators have been

developed. Among such generators are the Nuc-PLoc webserver (Shen & Chou, 2007), PseAAC webserver (Shen & Chou, 2008), GPCR-GIA webserver (Lin, et al., 2009), PseAAC-Builder (Du, et al., 2012), propy (Cao, et al., 2013), and PseAAC-General (Du, et al., 2014).

PseAAC generation often involves large-scale data processing. As such, computation time is of the essence. Software-based PseAAC generators have limitation in terms of further reducing the computation time due to the sequentially-executed nature of software codes. As such, it is of interest to this research to explore a faster alternative solution to generating PseAAC by proposing a hardware-based generator that has improved computation speed over the software-based version.

1.2 Problem Statement

PseAAC is one of the most widely used model in protein prediction. Together with machine learning algorithm such as Covariance Discriminant (CD) (Xu, et al., 2013), Fuzzy K-Nearest Neighbor (K-NN) (Xiao, et al., 2013) and Support Vector Machine (SVM) (Kumar, et al., 2015) among others (Qiu, et al., 2014), PseAAC has been used in a number of bioinformatics applications involving large-scale dataset processing.

Bioinformatics applications are usually software-based and developed to run on general purpose computer. Computer programs typically operate in sequential manner, with lines of codes executed serially. The speed of running the program is dependent on the processor clock speed. In today's computer, the processor is capable of running at the gigahertz range, making it a popular choice for running bioinformatics applications. In

spite of this, there is growing interest in exploring alternatives to computers in search of higher computation speed. One such alternative is programmable hardware (Aluru & Jammula, 2014).

Programmable hardware such as Field Programmable Gate Array (FPGA) has steadily gained prominence in bioinformatics (Dollas, 2014). One of the main advantage of FPGA is its capability in parallel processing at all levels. As protein prediction usually involves large number of protein sequences, it is highly desired for the process to be faster. One area of protein prediction that can take advantage of improved computation speed is the generation of PseAAC.

Two of the most prominent applications for generating PseAAC are the PseAAC webserver (Shen & Chou, 2008) and PseAAC-Builder (Du, et al., 2012). These applications run on general purpose computer. The webserver requires internet connection in order to generate PseAAC. As such, the duration required to obtain results may vary according to internet connection speed. This can be a disadvantage to user with slower connection. The PseAAC-Builder, on the other hand, offers a standalone package that installs the computation engine and supporting frameworks on a local computer. It being a computer software, however, means the execution of codes are performed sequentially. As such, its potential for improvement in computation speed is limited. The FPGA-based PseAAC generator can overcome the limitations of these applications as it does not require internet connection and has the potential for faster computation speed through parallel processing. Motivated by these advantages, this research proposes an architecture for an FPGA-based accelerator for the generation of PseAAC.

1.3 Objectives

This research has the following objectives:

- To propose the PseAAC generator's Register Transfer Level (RTL) design and implement it on hardware using FPGA.
- To establish accuracy and performance speed up of the FPGA-based PseAAC generator by comparing its results and computation time to a software-based generator that is developed using the Perl programming language. Both generators are implemented with the same PseAAC algorithm. The Perl-based generator's accuracy is also measured against the PseAAC webserver.

1.4 Research Scope

The scope of the research covers the following area:

1) Review of the two most common PseAAC modes; Chou's Type 1 and Type 2 PseAAC, which are supported by most bioinformatics applications such as the Nuc-PLoc webserver (Shen & Chou, 2007), PseAAC webserver (Shen & Chou, 2008), GPCR-GIA webserver (Lin, et al., 2009), PseAAC-Builder (Du, et al., 2012), propy (Cao, et al., 2013), and PseAAC-General (Du, et al., 2014).
2) PseAAC generator implementation on software platform using Perl programming language that runs on a general purpose computer. This version of the generator is used in performance comparison to the FPGA-based version. For fair comparison, both Perl-based and FPGA-based versions are developed using the same PseAAC algorithm.

3) Accuracy measurement of the Perl-based generator benchmarked by the PseAAC webserver (Shen & Chou, 2008). The webserver is chosen because it is also the benchmark tool of choice by the PseAAC-Builder (Du, et al., 2012).
4) RTL architecture development of the FPGA-based PseAAC generator using Verilog-Hardware Description Language (HDL). The design is developed using the Altera Quartus II Web Edition design software and realized using the Cyclone III FPGA.
5) Computation time evaluation of the FPGA-based PseAAC generator. The results are compared to the Perl-based version to establish performance gain. The accuracy of the FPGA-based version will also be compared to the Perl-based version.

1.5 Research Contribution

This research contributes to run time improvement of PseAAC generation by proposing an FPGA-based architecture which has improved computation speed over a software version that runs on a general purpose computer. The software version is developed using Perl programming language. The improvement is achieved by capitalizing on the parallel processing capability of FPGA.

1.6 Thesis Outline

The content of this thesis is arranged into five chapters.

In Chapter 1, the background of the research is reviewed. The problem statement and objectives are defined to establish clear goals of the research. The scope of the research is also outlined to narrow down the area of focus. Contribution by the research is also discussed.

In Chapter 2, the outcome of a comprehensive literature review is deliberated. The review covers various subjects related to the research such as the challenges of protein prediction post-genomic age, protein sequence modeling such as sequential and discrete modeling, the Type 1 and Type 2 PseAAC equations, bioinformatics applications related to the generation of PseAAC and the role of programmable hardware in bioinformatics.

In Chapter 3, the methodology of the research is charted. The four development stages of the research are discussed in detail. Functional specification of the proposed PseAAC generator is assessed in the first stage. The development strategy of the Perl-based generator is outlined in the second stage. It includes discussion on the PseAAC computation flows and accuracy evaluation of the Perl-based generator. In the third stage, development strategy of the FPGA-based generator is discussed. The RTL architecture of the generator is also proposed and its various modules are explained in detail. In the fourth stage, accuracy and computation speed measurements of the Perl-based and FPGA-based generator are discussed.

In Chapter 4, results of various evaluations on the Perl-based and FPGA-based generator are presented and discussed. The evaluations include accuracy measurements

of the Perl-based and FPGA-based generator, compilation and simulation results of the RTL design and computation time comparison between the Perl-based and FPGA-based generator. The reasons and implications of each result is explored to gain insights into the proposed software and hardware generator's performance.

In Chapter 5, the overall summary of the research is made. The limitation of the proposed FPGA-based generator is also highlighted and recommendations to improve and expand the performance of the generator are discussed.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter discusses a variety of research subjects related to PseAAC. It reviews the challenges faced by the research community in the post genomic age and protein modeling techniques that have been developed for protein prediction. PseAAC is a modeling technique developed to improve prediction quality. This technique has been widely used in many bioinformatics applications. It has been adapted into several different modes, each tailored to specific requirements to further enhance the accuracy. In the ensuing sections, two of the most common PseAAC modes, Type 1 and Type 2, will be examined. These modes are also popularly known as the basic PseAAC and amphiphilic PseAAC respectively.

In the wake of PseAAC's popularity, several bioinformatics applications have been developed to process large set of protein sequences into PseAAC. These applications come in various forms such as webservers and standalone software. The emergence of programmable hardware such as FPGA as a faster solution for bioinformatics applications will also be reviewed. The potential for improved computation speed by capitalizing on the parallel processing ability of FPGA has led to its increased role in bioinformatics.

2.2 Protein and protein sequences

Protein is a macromolecule formed by one or more chains of amino acid residues. It is one of the building blocks of life that serves a variety of biochemical functions in a living organism (Singh, et al., 2012). For example, proteins in a human body act as support structure for cells and bind the body together, as enzymes to store and release energy, as transporters to move molecules within the body, as hormones to regulate bodily activity and as antibodies against infections.

A polypeptide is a chain of amino acid residues. Each residue in a polypeptide is bonded to an adjacent residue linearly by a peptide bond. An illustration of a polypeptide is shown in Figure 2.1. Protein sequence is the arrangement of the amino acid residues in a polypeptide. The arrangement is determined by the genetic information within the Deoxyribonucleic Acid (DNA) of a particular polypeptide. Generally, all residues can be categorized into 20 native amino acid types. Table 2.1 lists the 20 native amino acids and their three- and one-letter codes. Each amino acid residue shown in Figure 2.1 is formed by one of the 20 native amino acids listed in Table 2.1.

Protein sequencing is the process of characterizing the sequence arrangement of amino acids in a protein. The sequence of a protein can be determined by techniques such

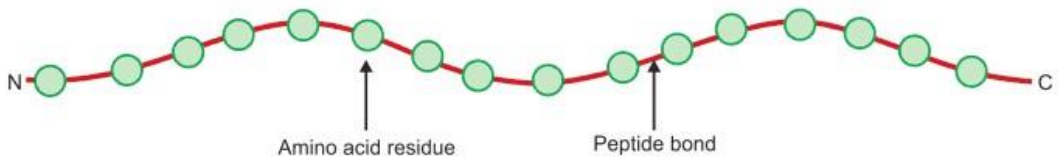

Figure 2.1: Schematic diagram of a polypeptide (Naik, 2012)