GENERALISATIONS OF SPLICING LANGUAGES FROM DEOXYRIBONUCLEIC ACID SPLICING SYSTEMS

NURUL IZZATY BINTI ISMAIL

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy

Faculty of Science
Universiti Teknologi Malaysia

JULY 2022

DEDICATION

To my parents, husband, and family, with love.

ACKNOWLEDGEMENT

All praises belong to Allah s.w.t for giving me His blessings and fulfilling my dream to complete this PhD study.

Foremost, I would like to express my most sincere appreciation to my main supervisor, Assoc. Prof. Dr. Fong Wan Heng, for her support, encouragement, and supervision during this study. I am also eternally thankful to my co-supervisor, Prof. Dr. Nor Haniza Sarmin, for her guidance, inspiration, and motivation. I truly appreciate their suggestions and valuable time in the preparation of this thesis.

Many thanks to Dr. Mohd Firdaus Abdul-Wahab from the Department of Biosciences, Faculty of Science, Universiti Teknologi Malaysia and his research assistant, Eka Latiffah Nadia Dzulkarnain, who assisted me in handling experimental works at Proteomics Laboratory at the Department of Biosciences, Faculty of Science, Universiti Teknologi Malaysia for ten months.

I would also like to acknowledge Universiti Teknologi Malaysia for funding my study for six semesters through Zamalah Scholarship.

My special gratitude goes to my parents, Ismail Samad and Haslina Abdul Rahim, for their unconditional love, continued prayers, and effort in bringing out the best in me to become a better person. To all my family members and friends, I am deeply grateful to them for being a part of this journey through thick and thin.

Thank you very much.

ABSTRACT

The mathematical modelling of recombinant deoxyribonucleic acid (DNA) utilises formal language theory which integrates the areas of applied discrete mathematics, theoretical computer science, and linguistics. In splicing systems, the presence of restriction enzymes allows DNA molecules to be cleaved and recombined to generate a new set of molecules, known as a splicing language, which can be analysed using formal language theory. Previous research on DNA splicing systems with different restriction enzymes has led to various splicing languages. However, these splicing languages are specific to the respective enzymes. In this research, the splicing languages are generalised based on the sequence of restriction enzymes which is either a palindromic sequence or a non-palindromic sequence. A palindromic sequence is a recognition sequence that reads the same way both forward and backward. Then, the splicing languages from the respective splicing systems are reduced to simple splicing systems via homomorphism. In addition, the models of splicing systems are verified through laboratory experiments to validate the theoretical results from these generalisations. Lastly, algorithms and also a graphical user interface (GUI) for splicing systems are developed using C++ visual programming to generate all splicing languages from the splicing systems involving palindromic or non-palindromic restriction enzymes. The results of this research include automata for the generalisations of splicing languages in simple splicing systems and the GUI for the computation of splicing languages. Moreover, the resulting molecules that depict the generalised splicing languages are documented in polyacrylamide gel electrophoresis (PAGE) gel photos as obtained from the experiments. The models presented in this research contribute to the advancement of DNA computing through the generalisations of splicing languages.

ABSTRAK

Pemodelan matematik asid deoksiribonukleik (DNA) rekombinan menggunakan teori bahasa formal yang mengintegrasikan bidang matematik diskret gunaan, sains komputer berteori, dan linguistik. Dalam sistem hiris-cantum, kehadiran enzim pembatasan membenarkan molekul-molekul DNA dipotong dan digabung semula untuk menghasilkan set molekul yang baharu, dikenali sebagai bahasa hiris-cantum yang boleh dianalisis menggunakan teori bahasa formal. Kajian terhadap sistem hiris-cantum DNA yang melibatkan pelbagai enzim pembatasan sebelum ini telah menghasilkan pelbagai bahasa hiris-cantum. Namun begitu, bahasa hiris-cantum ini adalah khusus kepada enzim pembatasan tertentu. Dalam kajian ini, bahasa hiris-cantum diitlakan berdasarkan jujukan enzim pembatasan iaitu jujukan palindromik atau jujukan tak palindromik. Jujukan palindromik adalah jujukan pengecaman yang dibaca sama ke hadapan dan ke belakang. Seterusnya, bahasa hiris-cantum daripada sistem hiris-cantum berkenaan diringkaskan kepada sistem hiris-cantum ringkas melalui homomorfisme. Tambahan pula, model sistem hiris-cantum ditentusahkan melalui uji kaji makmal untuk mengesahkan hasil teori daripada pengitlakan tersebut. Akhir sekali, algoritma dan antara muka grafik pengguna (GUI) bagi sistem hiris-cantum dibangunkan dengan menggunakan pengaturcaraan visual C++ untuk menjana semua bahasa hiris-cantum daripada sistem hiris-cantum yang melibatkan enzim pembatasan palindromik atau tak palindromik. Hasil kajian ini merangkumi automata bagi pengitlakan bahasa hiris-cantum dalam sistem hiris-cantum ringkas dan GUI bagi pengiraan bahasa hiris-cantum. Tambahan lagi, molekul terhasil yang menggambarkan bahasa hiris-cantum teritlak didokumenkan dalam gambar gel elektroforesis poliakrilamida (PAGE) seperti yang diperolehi daripada uji kaji ini. Model yang dibentangkan dalam kajian ini menyumbang kepada kemajuan pengkomputeran DNA melalui pengitlakan bahasa hiris-cantum.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CPP	-	C plus plus
DNA	-	Deoxyribonucleic acid
dsDNA	-	Double-stranded DNA
DFA	-	Deterministic finite automata
EC	-	Enzyme Commission
GUI	-	Graphical user interface
NEB	-	New England Biolabs
PAGE	-	Polyacrylamide gel electrophoresis
PCR	-	Polymerase chain reaction
Y-G	-	Yusof-Goode

LIST OF SYMBOLS

A	-	Adenine
A	-	Alphabet
M	-	Automaton
$\cdot$	-	Concatenation
C	-	Cytosine
$\in$	-	Element of
$\emptyset$	-	Empty set
λ	-	Empty string
$=$	-	Equal to
$\geq$	-	Greater than or equal to
G		Guanine
OH	-	Hydroxyl
q_0	-	Initial state
Σ	-	Input alphabet
L	-	Language
$L(S)$	-	Language generated by a splicing system S
$\leq$	-	Less than or equal to
$\uparrow$	-	Lower cutting sites of the restriction enzymes.
$-$	-	Minus
$\notin$	-	Not element of
$\neq$	-	Not equal to
( ), { }	-	Parentheses
r	-	Regular expression
B	-	Set of all patterns associated with enzymes that either produce $5'$ overhangs or blunt ends
C	-	Set of all patterns associated with enzymes that produce $3'$ overhangs
F	-	Set of final states
I	-	Set of initial strings/states
$\mathbb{Z}^+$	-	Set of positive integers

R	-	Set of splicing rules
Q	-	Set of states
A^*	-	Set of strings of concatenating zero or more symbols from an alphabet A
A^+	-	Set of strings of symbols from an alphabet A without zero
S_k	-	Splicing language families
r	-	Splicing rule
σ	-	Splicing scheme
S	-	Splicing system
*	-	Star-closure
T	-	Thymine
δ	-	Transition function
+, ∪	-	Union
↓	-	Upper cutting sites of the restriction enzymes.

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Introduction

The field of deoxyribonucleic acid (DNA) computing emerged from the idea of performing computations with biological materials, which was proposed by Feynman [1] in 1961. The major function of DNA as part of the biological materials incorporated in the computation gives considerable interest to many researchers from molecular biology, mathematics, computer science, and engineering [2]. The advancement in DNA computing was realised in Adleman's experiment on solving the Hamiltonian path problem based on two features: the massive parallelism of DNA strands and the Watson-Crick complementarity [3]. In massive parallelism, computation on DNA is derived as a DNA-based mechanism for solving intractable problems [4] where the DNA strand is a chain composed of nucleotides. Meanwhile for Watson-Crick complementarity, each double-stranded DNA (dsDNA) molecule appears in a double helix structure of DNA which is formed by nitrogenous base pairing [5]. While the results of Adleman's experiment serve as a practical application in DNA computing, research in this area also involves the theory, methods, and algorithm designs concerned with the use of DNA. In wet-lab experiments, the presence of restriction enzymes and a DNA ligase enable DNA molecules to be cleaved and recombined to form further molecules. The mathematical formalism for the molecular process of recombinant DNA under the given enzymatic activities that utilises formal language theory is called a DNA splicing system [6].

Formal language theory is a study spurred on by the core of theoretical computer science and discrete mathematics in the theory of computation [7]. A formal language is a set of strings (or words) of symbols derived from an alphabet under some formation rules. Basically, formal languages are composed of symbols in mathematical notations, numeral systems, logic operations, and programming

languages. Unlike natural languages (such as English and Bahasa Melayu) commonly used for communication in daily life, Latin alphabets with the upper and lower case letters serve as symbols to create or spell words. In making proper and complete sentences, the words are arranged according to the basics of grammar. For formal languages, the strings can also be structured by the syntax of grammar (also known as formal grammar) which was first proposed by Chomsky [8]. A formal grammar consists of a set of productions with terminal and non-terminal symbols in which the terminal symbols are the elements of an alphabet in the respective formal language; while the non-terminal symbols are placeholder that can be replaced with the terminal symbols to form the strings based on the productions.

Formal language theory can be associated with automata theory within the framework of the theory of computation, by which automata are described as the graphical representations of formal languages in state diagrams [7]. An automaton is made up of a series of states and a set of input symbols. By moving through the states, strings concatenated from the input symbols are produced by the automaton to generate a language. Based on the Chomsky hierarchy introduced in [8], the automata can be categorised by the classes of formal languages, which evolved into different types of grammars. To establish the relation between automata theory and grammar, the grammar formalism is used in designing the automata. The non-terminal symbols in formal grammar represent the states where the transitions between the states are determined by the productions. On the other hand, the strings of the input symbols are formed by the sequences of the terminal symbols. Hence, the language generated by the grammar can be recognised by the automata. The concepts of automata can be applied in splicing systems where languages that result from the splicing systems can be visualised in automata diagrams using grammars. The relation between splicing systems, automata, and grammars had been discussed in [9]. Besides, the study of splicing system under a systematic transition diagram was performed by constructing automata that recognise the languages generated by splicing systems via grammar [10].

In the modelling of DNA splicing systems, the base pairs, DNA molecules, and restriction enzymes are denoted as symbols, strings, and rules respectively by using formal language theory. The language resulting from a DNA splicing system is known

as a splicing language [6]. Splicing languages can be generated as a result of many variants of splicing systems, distinguished by different rules taken from restriction enzymes. Manual computations to obtain the splicing languages generated by the respective DNA splicing systems are often time-consuming. The purpose of this research is to determine the generalised splicing languages in order to shorten the computations. Hence, generalisations of splicing languages refer to the resulting dsDNA strings that are summarised from the respective splicing systems using formal language theory.

In the first part of this research, generalisations of splicing languages from DNA splicing systems are determined and presented based on the rules. A vast majority of previous research on DNA splicing systems, splicing languages, and rules is based on dsDNA molecules in the form of single strings. Thus in the second part of this research, the splicing systems are reduced to simple splicing systems in which the generalised splicing languages for the latter are also obtained. For graphical visualisation of the generalised splicing languages, the respective splicing systems are constructed using concepts in automata theory and grammar. Next in the third part of this research, wet-lab experiments for the splicing system are carried out to verify the actual results from the experiments with these generalisations. Lastly, some algorithms are built from these generalisations to develop a graphical user interface for generating the splicing languages depending on the inputs of DNA molecule and restriction enzymes.

1.2 Research Background

In 1987, Head [6] introduced the first theoretical model of a DNA splicing system that simulates the process of recombinant DNA under the framework of formal language theory. An early experimental study of producing recombinant DNA molecules with the presence of enzymes was conducted by Cohen et al. [11] in 1973. The process that occurs in the splicing system, known as a splicing operation, takes place by two types of enzymes: restriction enzymes and a DNA ligase. Each restriction enzyme has a recognition sequence that detects a specific nucleotide sequence in DNA

molecules. In the process, the molecules are cut by the restriction enzymes at the recognition sites and are then rejoined by the ligase to form the same or other molecules. The splicing system is analysed using formal language theory whereby the set of molecules (or DNA strings) generated throughout the process is called a splicing language.

For structural purposes, DNA molecules consist of the upper and lower strands of nucleotide sequences where both strands are complementary to each other. The sequences of molecules can read in two ways; forwards and backwards (if inverted 180^{o}) due to the complementary nature of the two strands [12]. In the context of word combinatorics, a string that reads the same forwards and backwards is known as a palindrome [13]. As mentioned in the first paper on splicing system by Head [6], the definition of a palindrome was presented in terms of formal language theory concerning the sequence effects of DNA structures. In 2012, Yusof [14] defined a palindromic string for dsDNA molecules in DNA splicing. Besides, a palindromic sequence is a recognition sequence that is recognised by certain restriction enzymes within DNA molecules if the sequence of the upper strand matches the sequence of the lower strand when read from backwards [15]. In DNA splicing systems, restriction enzymes act as rules that play a significant part in generating all potential DNA strings. For simplicity in notation, a restriction enzyme with a palindromic sequence is denoted by a palindromic rule, whereas a non-palindromic rule indicates a restriction enzyme with a non-palindromic sequence.

According to Head's splicing model, the rules are represented by triples that indicate the cleavage patterns of restriction enzymes [6]. For each triple, the string of a rule is termed a site that comprises left context, crossing, and right context [16]. From the molecular aspect, DNA molecules are cut in different ways depending on the cleavage patterns in which $5'$ overhang, $3'$ overhang, and blunt end are produced by restriction enzymes [17]. The $5'$ overhang and $3'$ overhang are sticky ends made by staggered cuts of the restriction enzymes on the molecules, while the restriction enzymes generate the blunt end from a straight cut down through both strands of the molecules. By referring to the definition of a splicing system in [6], Head's splicing model consists of a set of initial DNA strings and two sets of triples over an alphabet,

where the alphabet is a set of symbols derived from DNA base pairs. Those two sets of triples are classified by left and right patterns of rules associated with restriction enzymes that produce $5'$ overhang (or blunt end) and $3'$ overhang, respectively. The splicing rules for Head's splicing model show that the crossings of triples correspond to the sticky ends since sticky-end ligations are specific to the dsDNA formation based on the base pairing. However, for blunt ends, the crossings are written as empty strings because blunt ended molecules can be ligated without involving the base pairing. Hence, the notation of splicing rules is designed to preserve the natural function of restriction enzymes towards DNA.

Throughout the years, the study of DNA and enzymes related to formal language theory has rapidly attracted researchers' interest, which resulted in various extended models of splicing systems with different forms of notations for splicing rules. In 1996, Paun [18] introduced a new formalism for splicing system indicated as a pair known as a splicing scheme. The pair contains an alphabet and a set of rules in the simplified form that only concerns the left and right contexts. Based on Paun's splicing operation, the resulting strings are linked independently since no crossing is involved in this splicing system. Although Bonizzoni et al. [19] claimed that the approach is theoretically powerful due to the effective links between the strings, Paun's splicing model does not work biologically without crossings to imitate the enzymatic effects on recombinant DNA molecules. Then in that year also, another splicing scheme was presented by Pixton [20] to define a set of rules wherein the splicing model performs by a substituting operation that allows two strings to be attached by replacing a site with a given string. Since the substituting operation applied in Pixton's splicing model is in opposition to the basic splicing operation, Goode and Pixton [21] in 2004 proposed a new splicing model that resembles the actual behaviour of recombinant DNA molecules through the splicing process. However, the notation for splicing rules tends to disadvantage this splicing model in identifying the left and right patterns of rules. In 2013, a theoretical analysis on Head, Paun, Pixton and Goode-Pixton's splicing models was made by Yusof et al. [22] to investigate the effectiveness of the splicing performance from biological perspectives. As a result, a Yusof-Goode ($Y - G$) splicing system was proposed with a new notation for writing splicing rules in double of triples based on the specified patterns. In 2014,

the concept of a fuzzy splicing system was also developed by Karimi et al. [23] as an extension of Paun's splicing model to enhance its computational power with respect to the fuzzy effect on the splicing operation.

Head's splicing model has been used to define several kinds of splicing systems such as null-context and simple splicing systems. A null-context splicing system was introduced by Head [6] where the splicing rules focus on crossings only whereby the left and right contexts are denoted as empty strings. Additionally, the splicing rules of the null-context splicing system were extended to a simple splicing system which was developed by Mateescu et al. [24]. In simple splicing systems, a simplified splicing rule representation is deduced from the notation of rules for the null-context splicing system. In order to verify the existence of a splicing language, the laboratory validation of Head's splicing model was done by Laun and Reddy [25], which contributes to the first wet-lab experiment for a biological (or wet) splicing system. The experiment was designed in an attempt to analyse the expected recombinant DNA molecules as predicted by the theoretical model. Inspired by the wet splicing system, a few experimental works on other splicing models and splicing languages have been conducted, which are shown in [26-29].

In a nutshell, research in this area has progressed remarkably in the past three decades due to the advent and interdisciplinary studies of theoretical splicing systems and their splicing languages with regard to the splicing operations and rules. This research concerns the reliability of splicing languages that result from DNA splicing systems, i.e., the splicing languages can be generalised depending on the splicing rules of the respective splicing systems by using Head's approach.

1.3 Problem Statement

Over the years, various types of splicing systems have been introduced with different operations, notations and rules. Splicing languages generated by the respective splicing systems are commonly obtained by manual computations which are time-consuming. There is no observable study on the shortening of computations

in splicing languages. Hence, this research focuses on generalising splicing languages in order to summarise all resulting dsDNA strings from different cases of DNA splicing systems involving palindromic and non-palindromic rules.

The generalised splicing languages are presented as sets of dsDNA strings to preserve the nature of recombinant behaviour through splicing operations. Based on previous research, the modelling of DNA splicing systems concerned on writing dsDNA strings as single strings. In this research, the generalised splicing languages are simplified using the arbitrary small-letter dsDNA symbols, which can be recognised in simple splicing systems. Also, since DNA molecules have long nucleotide sequences made up of numerous base pairs, then the graphical representations for the splicing systems are proposed to visualise the generalised splicing languages through automata.

Based on the generalisations of splicing languages which are obtained mathematically, the laboratory verification of these generalisations is performed to enhance the reliability of these theoretical results in an experimental manner. Thus in this research, the molecular aspects of the generalisations of splicing languages are investigated through wet-lab experiments.

Lastly, from the literature, there exists no recognisable invention which serves as an efficient alternative for the manual computation of splicing languages. Hence, a GUI for DNA splicing systems is created as an application program to generate splicing languages promptly based on the given initial DNA molecule and restriction enzymes.

The following questions are answered in this research:

1. What are the generalised splicing languages that result from DNA splicing systems with palindromic and non-palindromic rules?

2. How to graphically represent the corresponding generalised splicing system through automata?

3. How to perform wet-lab experiments for laboratory verification of the generalisations of splicing languages?

4. How to design algorithms and a programming code for the corresponding splicing systems in a graphical user interface?

1.4 Research Objectives

The objectives of the research are:

1. to generalise the resulting splicing languages from different DNA splicing systems with palindromic and non-palindromic rules.

2. to reduce the generalised splicing languages from the splicing systems to simple splicing systems.

3. to construct graphical representations of the generalised splicing languages in simple splicing systems via automata.

4. to verify the theoretical results for the generalisations of splicing languages through wet-lab experiments.

5. to design algorithms and also develop a graphical user interface for DNA splicing systems in a C++ programming language.

1.5 Scope of the Research

In this research, the splicing languages are generalised using Head's splicing model to interpret the process of cutting and pasting DNA using the splicing operations and also the properties of restriction enzymes as splicing rules. The generalisations of splicing languages are classified in the corresponding DNA splicing systems based on palindromic and non-palindromic rules, the number of cutting sites, and crossings of the rules. This research presents the generalisations of splicing languages that are

generated by the splicing systems involving one or two rules for one or two non-overlapping cutting sites with the same and different crossings. Besides, the wet- lab experiments in this research involve restriction enzymes *Cvi*QI and *Aci*I, and T4 DNA ligase in the laboratory procedures in order to verify the generalisations of splicing languages through the experiments.

1.6 Significance of the Research

Mathematical research in the field of molecular biology has generated new discoveries that benefit the development of applications in DNA computing. DNA splicing system is a mathematical formalism for the biological phenomenon through splicing operations which is inspired by DNA computing models. In this research, DNA splicing systems provide the generalisations of splicing languages theoretically, which depict the results from wet-lab experiments that are often costly and time-consuming.

Furthermore, the graphical user interface for DNA splicing systems is beneficial for mathematicians and biologists in the advancement of recombinant DNA technology. Mathematicians involved in this area of research can utilise this interface to obtain the splicing languages from the respective splicing systems. Moreover, the process of recombinant DNA is quite familiar among biologists since it is one of the laboratory methods of genetic recombination. The GUI also benefits computer scientists to model the universal programmable DNA-based computers that rely on DNA code, since in DNA computers, data are stored in the base sequence of DNA instead of silicon chips, and the computer is used to record data for a living cell.

1.7 Research Methodology

This research has been carried out to achieve the objectives through five phases, as shown in the following.

Phase 1: Literature review on concepts of DNA, splicing systems, formal language theory, automata theory, and grammar.

This research begins with studying the structures of DNA and variants of splicing systems including the splicing languages. Then, the roles of restriction enzymes and DNA ligases in the splicing systems are also analysed, whereby the restriction enzymes and ligases mentioned in this research are obtained from the New England Biolabs (NEB) catalogue [30]. Furthermore, some basic concepts in formal language theory, automata theory, and grammar with their relations are explored.

Phase 2: Generalise the splicing languages resulting from DNA splicing systems.

The splicing languages generated by Head's splicing model are generalised according to the palindromic and non-palindromic rules, and the cutting sites and crossings of the rules. By using formal language theory, the generalised splicing languages are described as regular expressions to specify the patterns of the resulted dsDNA strings through the splicing operation. The generalisations of splicing languages are presented in theorems and lemmas which are proven using direct, induction, contradiction and contrapositive methods.

Phase 3: Construct graphical representations for the generalisations of splicing languages in simple splicing systems via automata.

The generalisations of splicing languages from the respective splicing systems, which are discussed in Phase 2, are simplified using the small-letter dsDNA symbols and then reduced to simple splicing systems via homomorphism. In order to construct the automata, the generalised splicing languages are deduced from grammars where the productions formed by the grammars correlate with the transitions between states in the automata. Then, the graphical representations of the languages generated by the grammars are visualised in automata diagrams to recognise the generalised splicing languages. The automata are constructed using a deterministic finite automaton (DFA) model that accepts the generalised splicing languages as regular languages which can be defined by the regular expressions.

Phase 4: Present the molecular aspects of the generalisations of splicing languages in wet splicing systems.

The experiments are designed by the modelling of wet splicing systems to determine the expected DNA molecules using the generalised splicing languages. There are three molecular methods involved in laboratory procedures to conduct the experiments: polymerase chain reaction (PCR) technique, the process of restriction enzyme digestion and DNA ligation, and the preparation of polyacrylamide gel electrophoresis (PAGE) gels. First, the PCR technique is performed to produce the initial DNA molecules, while the recombinant DNA molecules are generated through the process of restriction enzyme digestion and DNA ligation. Then, the molecules resulting from the experiment are visualised on the PAGE gels to verify the generalised splicing languages.

Phase 5: Develop algorithms and a graphical user interface (GUI) for DNA splicing systems.

The generalised splicing languages are applied in the algorithms which are designed by a series of steps to compute the splicing languages. Then, the algorithms are implemented in a C++ programming code. The GUI for DNA splicing systems is developed using Microsoft Visual Studio to execute the programming code and to design the interface for receiving inputs from users and for generating outputs. On the interface, the users are required to insert an initial DNA string and rules taken from any restriction enzymes. As the outputs of the GUI, the splicing languages that contain the resulting molecules are displayed on the interface based on the respective inputs entered by the users.

Next, the research framework for this thesis is illustrated in Figure 1.1, where those grey-coloured boxes correspond to the objectives of this research.

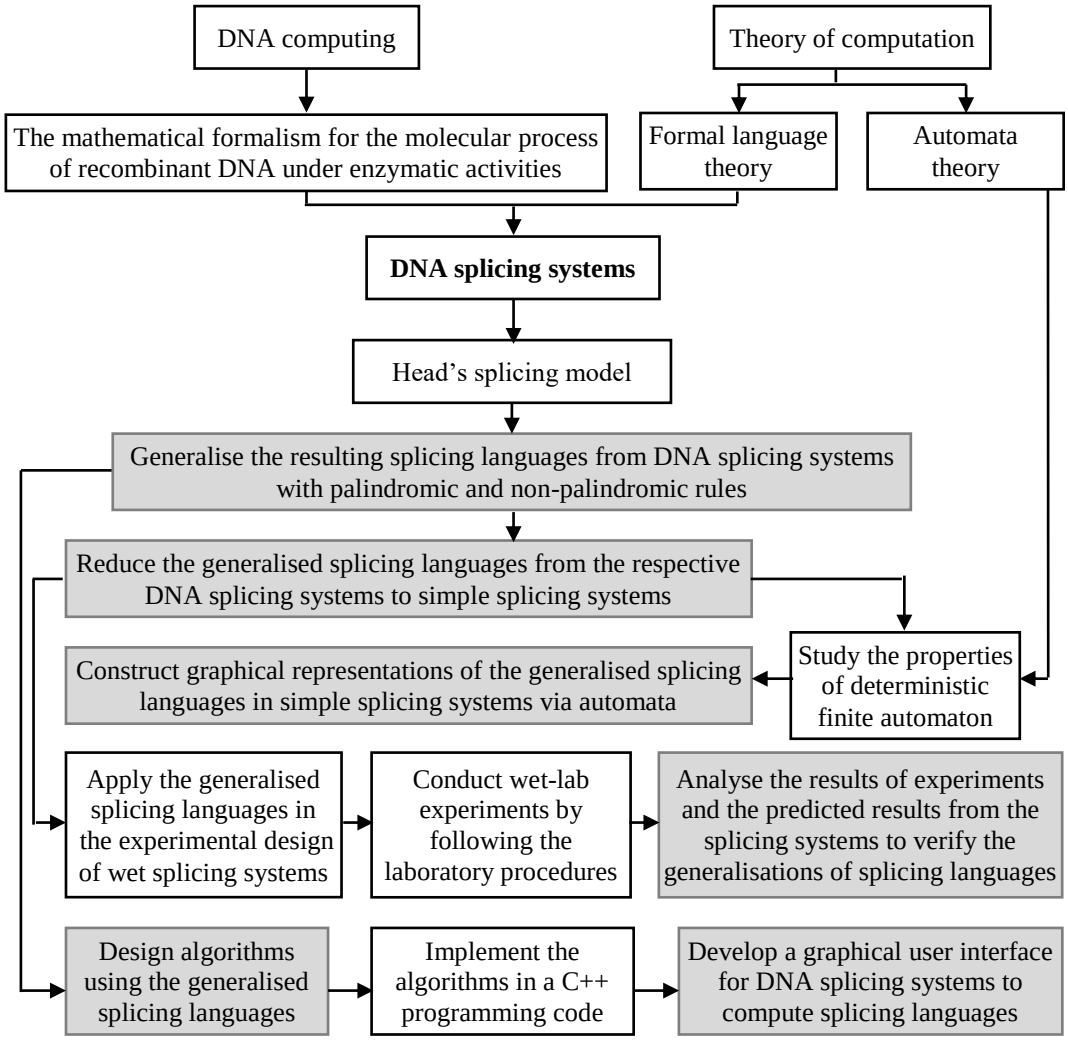

Figure 1.1 Research framework

1.8 Thesis Organisation

This thesis is divided into seven chapters. The first chapter gives a brief idea of this research which includes the research background, problem statement, research objectives, scope of the research, significance of the research, and research methodology.

The literature review on the structures of DNA and the historical background of DNA splicing systems are presented in Chapter 2. Besides, the properties of restriction enzymes with palindromic and non-palindromic sequences, and the function of DNA ligases in the splicing systems, are discussed. Moreover, some

concepts in formal language theory, automata theory, and grammar are given in this chapter. The preliminaries that are used throughout this research are also stated.

Chapter 3 presents the generalisation of splicing languages from DNA splicing systems with different rules as stated in the theorems. In this chapter, these generalisations are arranged into three sections based on the rules: palindromic rules, non-palindromic rules, and both palindromic and non-palindromic rules.

Chapter 4 begins by showing the relation between simple splicing systems, automata theory, and grammar. Then, the generalisations of splicing languages in simple splicing systems that are reduced from the DNA splicing systems, are given in propositions which are preceded by corollaries that simplify these generalisations using the small-letter dsDNA symbols. Finally, the graphical representations of the generalised splicing languages in the simple splicing systems via automata are presented as theorems.

In Chapter 5, the molecular aspects on the generalisations of splicing languages in wet splicing systems are discussed. The experimental design of the splicing systems is presented to determine the expected DNA molecules. Next, the laboratory procedures are given to demonstrate the steps of conducting the wet-lab experiments. Then, the actual results of the experiments are analysed to prove the existence of the expected molecules.

Chapter 6 is dedicated to the graphical user interface for DNA splicing systems in which the objectives of the programming code are stated as a brief overview of the purpose of GUI. In this chapter, the algorithms for the generalisations of splicing languages are presented as flowcharts to illustrate the programming procedure. Also, the instructions to use the interface are provided. Furthermore, several examples of DNA splicing systems are given to show how the GUI functions to generate the splicing languages as outputs.

Lastly, the whole research is summarised in Chapter 7. Besides, some suggestions for further research on the generalisations of splicing languages are also given in this chapter.

Figure 1.2 is shown to visualise the outline of the thesis.

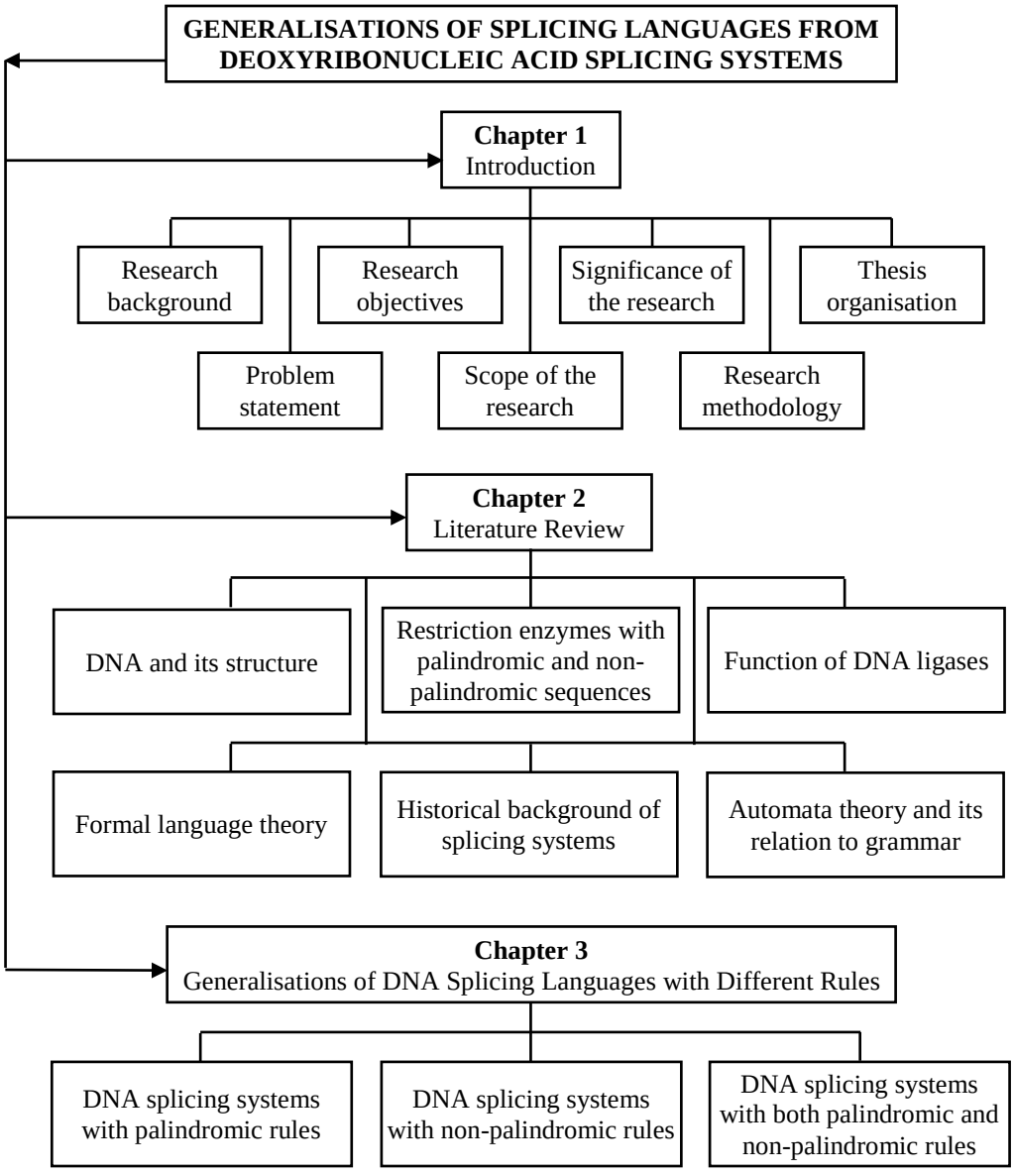

Figure 1.2 Thesis outline

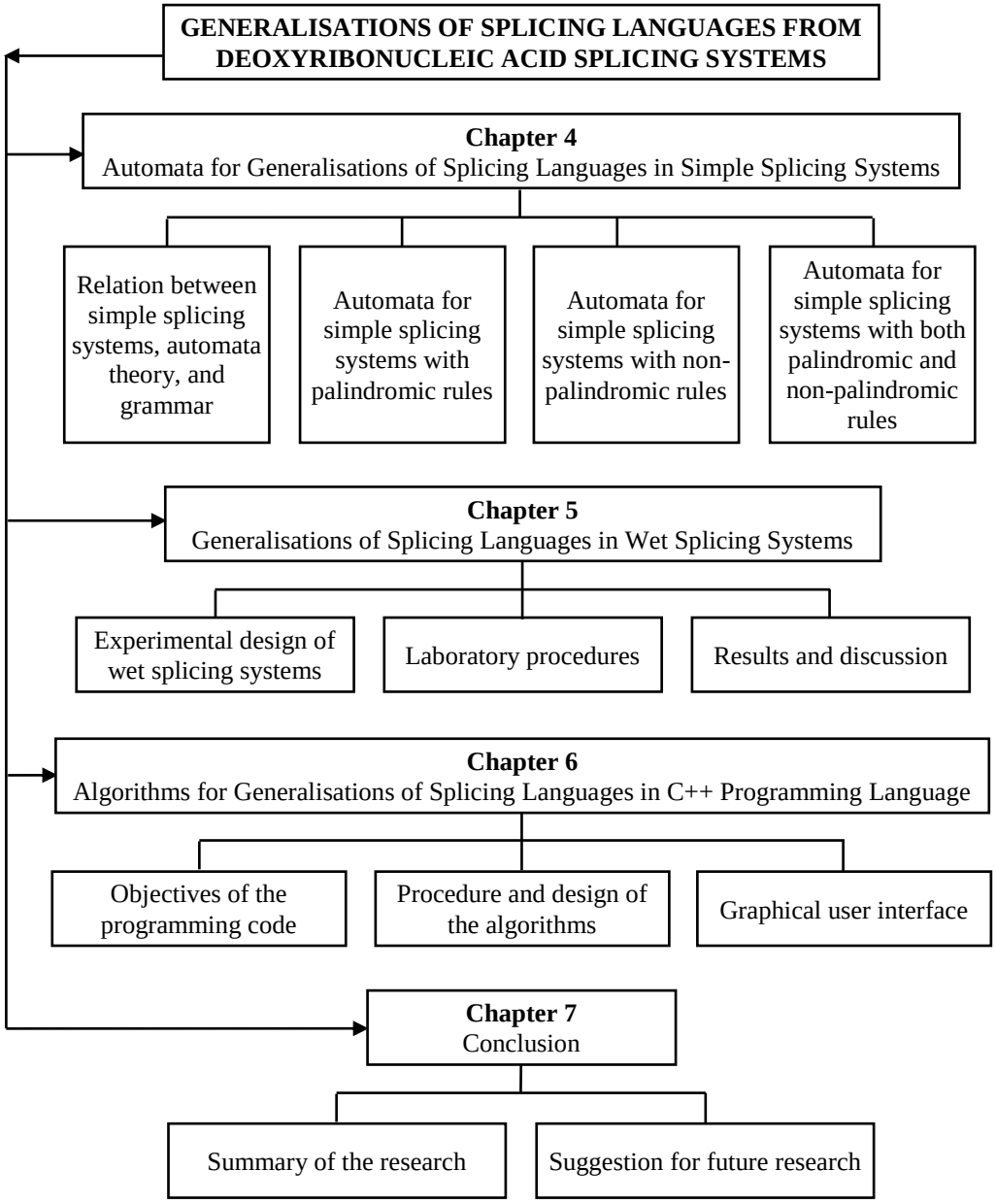

Figure 1.2 Thesis outline (cont.)

REFERENCES

1. Feynman, R. P. There's Plenty of Room at the Bottom, In: H.D. Gilbert ed. *Engineering and Science*. New York: Reinhold Publishing Company. 282–296; 1961.
2. Păun, G., Rozenberg, G., and Salomaa, A. *DNA Computing: New Computing Paradigms*. Germany: Springer -Verlag Berlin Heidelberg. 1998.
3. Adleman, L. M. Molecular Computation of Solutions to Combinatorial Problems. *Science*. 1994. 266(5187): 1021-1024.
4. Beaver, D. Computing with DNA. *Journal of Computational Biology*. 1995. 2(1): 1-7.
5. Watson, J. D. and Crick, F. H. C. The structure of DNA. *Proceedings of the Cold Spring Harbor Symposia on Quantitative Biology*. June 5-11, 1953. NY, United States: Cold Spring Harbor Laboratory Press. 1953. 123-131.
6. Head, T. Formal Language Theory and DNA: An Analysis of the Generative Capacity of Specific Recombinant Behaviors. *Bulletin of Mathematical Biology*. 1987. 49(6): 737-759.
7. Linz, P. *An Introduction to Formal Languages and Automata*. 4th ed. USA: Jones and Bartlett Publisher. 2006.
8. Chomsky, N. Three Models for the Description of Language. *IRE Transactions on Information Theory*. 1956. 2(3): 113-124.
9. Mohamad Jan, N., Fong, W. H., and Sarmin, N. H. Regular Languages, Regular Grammars and Automata in Splicing Systems. *Proceedings of the 20th National Symposium on Mathematical Sciences: Research in Mathematical Sciences: A Catalyst for Creativity and Innovation*. December 18–20, 2012. Melville, NY: AIP Publishing. 2013. 856-863.
10. Khairuddin, S., Ahmad, M. A., and Adzhar, N. Splicing System in Automata Theory: A Review. *Proceedings of the 2nd International Conference on Applied & Industrial Mathematics and Statistics*. 23–25 July 2019. UK: IOP Publishing. 2019. 012066 1-10.

11. Cohen, S. N., Chang, A. C. Y., Boyer, H. W., and Helling, R. B. Construction of Biologically Functional Bacterial Plasmids *In Vitro. Proceedings of the National Academy of Sciences*. 1973. 70(11): 3240-3244.
12. Watson, J. D., Baker, T. A., Bell, S. P., Gann, A., Levine, M., and Losick, R. *Molecular Biology of the Gene*. 7th ed. New York: Cold Spring Harbor Laboratory Press. 2014.
13. Tomohiro, I., Inenaga, S., and Takeda, M. Palindrome Pattern Matching. *Theoretical Computer Science*. 2013. 483: 162-170.
14. Yusof, Y. *DNA Splicing System Inspired by Bio Molecular Operations*. Ph.D. Thesis. Universiti Teknologi Malaysia; 2012.
15. Eun, H. M. *Enzymology Primer for Recombinant DNA Technology*. USA: Academic Press. 1996.
16. Head, T. Splicing Representations of Strictly Locally Testable Languages. *Discrete Applied Mathematics*. 1998. 87(1): 139-147.
17. Ijaz, S. and Haq, I. U. *Recombinant DNA Technology*. Newcastle upon Tyne, United Kingdom: Cambridge Scholars Publishing. 2019.
18. Păun, G. On the Splicing Operation. *Discrete Applied Mathematics*. 1996. 70(1): 57-79.
19. Bonizzoni, P., Ferretti, C., Mauri, G., and Zizza, R. Separating Some Splicing Models. *Information Processing Letters*. 2001. 79(6): 255-259.
20. Pixton, D. Regularity of Splicing Languages. *Discrete Applied Mathematics*. 1996. 69(1-2): 101-124.
21. Goode, E. and Pixton, D. Splicing to the Limit, In: N. Jonoska, G. Păun, and G. Rozenberg ed. *Aspects of Molecular Computing, Lecture Notes in Computer Science*. Germany: Springer-Verlag. 189-201; 2004.
22. Yusof, Y., Sarmin, N. H., Fong, W. H., Goode, T. E., and Ahmad, M. A. An Analysis of Four Variants of Splicing System. *Proceedings of the 20th National Symposium on Mathematical Sciences - Research in Mathematical Sciences: A Catalyst for Creativity and Innovation (SKSM 2012)*. December 18-20, 2012. Melville, NY: AIP Publishing. 2013. 888-895.
23. Karimi, F., Turaev, S., Sarmin, N. H., and Fong, W. H. Fuzzy Splicing Systems, In: D. Hwang, J.J. Jung, and N.T. Nguyen ed. *Computational Collective Intelligence. Technologies and Applications, ICCCI 2014, Lecture*

Notes in Computer Science. Cham, Switzerland: Springer International Publishing. 20-29; 2014.

24. Mateescu, A., Păun, G., Rozenberg, G., and Salomaa, A. Simple splicing systems. *Discrete Applied Mathematics*. 1998. 84(1-3): 145-163.
25. Laun, E. and Reddy, K. J. Wet Splicing Systems. *Proceedings of the 3rd DIMACS Workshop on DNA Based Computers*. June 23-25, 1997. Rhode Island, USA: American Mathematical Society. 1999. 73-84.
26. Fong, W. H. *Modelling of Splicing Systems using Formal Language Theory*. Ph.D. Thesis. Universiti Teknologi Malaysia; 2008.
27. Karimi, F. *Mathematical Modelling of Persistent Splicing Systems in DNA Computing*. Ph.D. Thesis. Universiti Teknologi Malaysia; 2013.
28. Yusof, Y., Lim, W. L., Goode, T. E., Sarmin, N. H., Fong, W. H., and Abd Wahab, M. F. Molecular Aspects of DNA Splicing System. *Proceedings of the International Conference on Mathematics, Engineering and Industrial Applications 2014 (ICoMEIA 2014)*. 28–30 May 2014. AIP Publishing. 2015. 050045 1-8.
29. Ahmad, M. A., Sarmin, N. H., Abdul Wahab, M. F., Fong, W. H., and Yusof, Y. Biomolecular Aspects of Second Order Limit Language. *Malaysian Journal of Fundamental and Applied Sciences*. 2018. 14(1): 15-19.
30. New England Biolabs Inc. *NEB 2019-20 Catalog & Technical Reference*. Ipswich, United States: Catalogue. 2019.
31. Giel-Pietraszuk, M., Hoffmann, M., Dolecka, S., Rychlewski, J., and Barciszewski, J. Palindromes in Proteins. *Journal of Protein Chemistry*. 2003. 22(2): 109-113.
32. Lamprea-Burgunder, E., Ludin, P., and Mäser, P. Species-Specific Typing of DNA Based on Palindrome Frequency Patterns. *DNA Research*. 2011. 18(2): 117-124.
33. Robinson, P. K. Enzymes: Principles and Biotechnological Applications. *Essays in Biochemistry*. 2015. 59: 1-41.
34. Palmer, T. and Bonner, P. L. *Enzymes: Biochemistry, Biotechnology, Clinical Chemistry*. 2nd ed. UK: Woodhead Publishing Limited. 2007.
35. Smith, H. O. and Nathans, D. A Suggested Nomenclature for Bacterial Host Modification and Restriction Systems and Their Enzymes. *Journal of Molecular Biology*. 1973. 81(3): 419-423.

36. Loukianov, E., Loukianova, T., and Periasamy, M. Cloning DNA Fragments between Two Adjacent/Overlapping Restriction Sites Using a "Positive Stuffer". *Biotechniques*. 1997. 22(5): 912-915.

37. Degtyarev, S. K., Belichenko, O. A., Lebedeva, N. A., Dedkov, V. S., and Abdurashitov, M. A. *Btr*I, a Novel Restriction Endonuclease, Recognises the Non-Palindromic Sequence 5′-CACGTC (-3/-3)-3′. *Nucleic Acids Research*. 2000. 28(11): e56.

38. McDonald, A. G., Boyce, S., and Tipton, K. F. Enzyme Classification and Nomenclature, In: *Encyclopedia of Life Sciences*. Chichester: John Wiley & Sons. 1-11; 2015.

39. Shinde, S. A., Chavhan, S. A., Sapkal, S. B., and Shrikhande, V. N. Recombinant DNA Technology and its Applications: A Review. *International Journal of MediPharm Research*. 2018. 4(2): 79-88.

40. Kleene, S. C. *Representation of Events in Nerve Nets and Finite Automata*. Santa Monica, CA: RAND Corporation. 1951.

41. Brejová, B., DiMarco, C., Vinar, T., Hidalgo, S. R., Holguin, G., and Patten, C., *Finding Patterns in Biological Sequences*, in *Unpublished project report for CS798G*. 2000: University of Waterloo, Fall.

42. Goode, E. and Pixton, D. Semi-Simple Splicing Systems, In: C. Martín-Vide and V. Mitrana ed. *Where Mathematics, Computer Science, Linguistics and Biology Meet*. Dordrecht: Springer. 343-352; 2001.

43. Laun, T. E. G. *Constants and Splicing Systems*. Ph.D. Thesis. State University of New York; 1999.

44. Selvarajoo, M., Fong, W. H., Sarmin, N. H., and Turaev, S. The Characteristics of Simple Splicing Languages over Permutation Groups. *Proceedings of the 27th National Symposium on Mathematical Sciences (SKSM27)*. November 26–27, 2019. Melville, NY: AIP Publishing. 2020. 060004 1-6.

45. Selvarajoo, M., Fong, W. H., Sarmin, N. H., and Turaev, S. The Properties of Semi-Simple Splicing System Over Alternating Group, A_3. *Proceedings of the International Conference on Mathematical Sciences (ICMS 2020)*. March 4-6, 2020. United Kingdom: IOP Publishing. 2021. 012001 1-8.

46. Turaev, S., Selvarajoo, M., Selamat, M. H., Sarmin, N. H., and Fong, W. H. Probabilistic Splicing Systems, In: N.T. Nguyen, et al. ed. *Advanced Methods*

for Computational Collective Intelligence. Berlin, Heidelberg: Springer-Verlag. 259-268; 2013.

47. Selvarajoo, M., Fong, W. H., Sarmin, N. H., and Turaev, S. Probabilistic Simple Splicing Systems. *Proceedings of the 3rd International Conference on Mathematical Sciences*. December 17–19, 2013. Melville, NY: AIP Publishing. 2014. 760-766.
48. Selvarajoo, M., Fong, W. H., Sarmin, N. H., and Turaev, S. Probabilistic Semi-Simple Splicing System and Its Characteristics. *Jurnal Teknologi*. 2013. 62(3): 21–26.
49. Turaev, S., Gan, Y. S., Othman, M., Sarmin, N. H., and Fong, W. H. Weighted Splicing Systems. *Proceedings of the 6th International Symposium on Intelligence Computation and Applications*. October 27-28, 2012. Berlin, Heidelberg: Springer-Verlag. 2012. 416-424.
50. Gan, Y. S., Fong, W. H., Sarmin, N. H., and Turaev, S. Weighted Simple and Semi-Simple Splicing Systems. *Malaysian Journal of Fundamental and Applied Sciences*. 2014. 10(4): 200-205.
51. Gatterdam, R. W. Algorithms for Splicing Systems. *SIAM Journal on Computing*. 1992. 21(3): 507-520.
52. Kobayashi, S. and Sakakibara, Y. Multiple Splicing Systems and the Universal Computability. *Theoretical Computer Science*. 2001. 264(1): 3-23.
53. Karimi, F., Sarmin, N. H., and Fong, W. H. Crossing-Preserved and Persistent Splicing Systems. *Proceedings of the 2011 Sixth International Conference on Bio-Inspired Computing: Theories and Applications*. 2011. New York: IEEE. 2011. 167-169.
54. Krithivasan, K., Chakaravarthy, V. T., and Rama, R. Array Splicing Systems, In: G. Păun and A. Salomaa ed. *New Trends in Formal Languages*. Berlin, Germany: Springer-Verlag. 346-365; 1997.
55. Chandra, P. H., Subramanian, K. G., and Thomas, D. G. Parallel Splicing on Images. *International Journal of Pattern Recognition and Artificial Intelligence*. 2004. 18(06): 1071-1091.
56. Sheena Christy, D. K., John Paul, P., and Thomas, D. G. *H*-Array System on Picture Languages with Array Rewriting Rules. *Proceedings of the 1st International Conference on Mathematical Techniques and Applications*.

January 30 - February 1, 2020. Melville, NY: AIP Publishing. 2020. 030002 1-18.

57. Mitrana, V., Petre, I., and Rogojin, V. Accepting Splicing Systems. *Theoretical Computer Science*. 2010. 411(25): 2414-2422.

58. Sheena Christy, D. K., Masilamani, V., Thomas, D. G., Nagar, A. K., and Robinson, T. Accepting *H*-array splicing systems and Their Properties. *Science and Technology*. 2018. 21(3): 298-309.

59. Berstel, J., Boasson, L., and Fagnot, I. Splicing Systems and the Chomsky Hierarchy. *Theoretical Computer Science*. 2012. 436: 2-22.

60. Amjad Basha, F., Kumaar, S. J., and Ahmed, F. Characterization of Central Splicing. *Gis Science Journal*. 2021. 8(5): 1052-1056.

61. Sheena Christy, D. K. and Thomas, D. G. Iso-Array Flat Splicing System. *Proceedings of the 11th National Conference on Mathematical Techniques and Applications*. January 11-12, 2019 Melville, NY: AIP Publishing. 2019. 020042 1-5.

62. Ahmad, M. A., Sarmin, N. H., Fong, W. H., and Yusof, Y. An Extension of First Order Limit Language. *Proceedings of the 3rd International Conference on Mathematical Sciences (ICMS3)*. December 17-19, 2013. Melville, NY: AIP Publishing. 2014. 627-631.

63. Ahmad, M. A., Sarmin, N. H., Fong, W. H., and Yusof, Y. Second Order Limit Language in Variants of Splicing System. *Proceedings of the 21th National Symposium on Mathematical Sciences (SKSM21)*. November 6-8, 2013. Melville, NY: AIP Publishing. 2014. 639-643.

64. Lim, W. L., Yusof, Y., and Mudaber, M. H. Modeling of DNA Single Stage Splicing Language via Yusof-Goode Approach: One String with Two Rules. *Proceedings of the 2nd ISM International Statictical Conference 2014 (ISM-II)*. August 12-14, 2014. Melville, NY: AIP Publishing. 2015. 695-699.

65. Mudaber, M. H. and Yusof, Y. Two Stages Splicing System. *Proceedings of the International Conference on Mathematics, Engineering & Industrial Applications 2014 (ICoMEIA 2014)*. May 28-30, 2014. Melville, NY: AIP Publishing. 2015. 050046.

66. O'rourke, T. C., O'neill, B. T., Cook, R. C., Taner, K. O., Synder, S. P., and Joyner, A. R. *Graphical User Interface*. US005349658A. 1994.

67. Freach, J. E., Moore, R., Fuiks, K. A., and Davis, K. D. *Graphical User Interface*. US006710788B1. 2004.
68. Lim, W. L., Yusof, Y., Rosli, N., and Mudaber, M. H. Modelling the Behaviour of Single Stage Splicing Language: A Yusof Goode Computational Approach. *Jurnal Teknologi*. 2015. 73(1): 135–138.
69. Selvarajoo, M. *Probabilistic Splicing and Sticker Systems in DNA Computing*. Ph.D. Thesis. Universiti Teknologi Malaysia; 2015.
70. Rosen, K. H. *Discrete Mathematics & Applications*. 8th ed. New York: McGraw-Hill Education. 2019.
71. Mealy, G. H. A Method for Synthesizing Sequential Circuits. *The Bell System Technical Journal*. 1955. 34(5): 1045-1079.
72. Moore, E. F. Gedanken-Experiments on Sequential Machines, In: C.E. Shannon and J. McCarthy ed. *Automata studies: Annals of Mathematical Studies*. Princeton, NJ: Princeton University Press. 129-153; 1956.
73. Kim, S. M. Computational Modeling for Genetic Splicing Systems. *SIAM Journal on Computing*. 1997. 26(5): 1284-1309.
74. Fong, W. H., Sarmin, N. H., and Ibrahim, Z. Recognition of Simple Splicing Systems using *SH*-Automaton. *Malaysian Journal of Fundamental and Applied Sciences*. 2008. 4(2): 337-342.
75. Ahmad, M. A. *Second Order Limit Language and Its Properties in Yusof-Goode Splicing System*. Ph.D. Thesis. Universiti Teknologi Malaysia; 2016.
76. Kari, L. and Ng, T. State Complexity of Simple Splicing. *Proceedings of the International Conference on Descriptional Complexity of Formal Systems*. July 17–19, 2019. Cham, Switzerland: Springer. 2019. 197-209.
77. Kari, L. and Ng, T. Descriptional Complexity of Semi-Simple Splicing Systems. *Proceedings of the 24th International Conference on Developments in Language Theory*. May 11–15, 2020. Cham, Switzerland: Springer. 2020. 150-163.
78. Erlich, H. A. Polymerase Chain Reaction. *Journal of Clinical Immunology*. 1989. 9(6): 437-447.
79. Dieffenbach, C. W., Lowe, T. M., and Dveksler, G. S. General Concepts for PCR Primer Design. *PCR Methods and Applications*. 1993. 3(3): S30-S37.

LIST OF PUBLICATIONS, COPYRIGHTS AND AWARDS

Indexed Journals

1. Fong, W. H. & **Ismail, N. I.** (2018). Generalisations of DNA Splicing Systems with One Palindromic Restriction Enzyme. *MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics*, 34(1), 59-71. https://doi.org/10.11113/matematika.v34.n1.1011. **(Indexed by ISI WOS)**

2. Fong, W. H., **Ismail, N. I.** & Sarmin, N. H. (2018). Computation of Splicing Languages from DNA Splicing System with One Palindromic Restriction Enzyme. *Malaysian Journal of Fundamental and Applied Sciences*, 14(2), 188-192. https://doi.org/10.11113/mjfas.v14n2.879. **(Indexed by ISI WOS)**

3. Fong, W. H., **Ismail, N. I.** & Sarmin, N. H. (2019). The Modelling of DNA Splicing Systems with Two Non-Palindromic Restriction Enzymes. *Indian Journal of Public Health Research & Development*, 10(6), 983-989. http://dx.doi.org/10.5958/0976-5506.2019.01515.8. **(Indexed by Scopus)**

4. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. (2019). DNA Splicing Systems with at Most Two Cutting Sites of a Non-Palindromic Restriction Enzyme. *MATEMATIKA: Malaysian Journal of Industrial and Applied Mathematics*, 35(2), 129-137. https://doi.org/10.11113/matematika.v35.n2.1108. **(Indexed by ISI WOS)**

5. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. (2019). Computation of Splicing Languages from DNA Splicing System based on Sequences of Restriction Enzymes. *International Journal of Engineering and Advanced Technology (IJEAT)*, 8(6S3), 31-42. https://doi.org/10.35940/ijeat.F1006.0986S319 **(Indexed by Scopus)**

6. Fong, W. H., **Ismail, N. I.** & Sarmin, N. H. (2019). Automata for DNA Splicing Languages with Palindromic and Non-Palindromic Restriction Enzymes using Grammars. *MATEMATIKA: Malaysian Journal of Industrial*

and Applied Mathematics, 35, 1-14. https://doi.org/10.11113/matematika.v35.n4.1260 **(Indexed by ISI WOS)**

7. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. (2020). Automata for DNA Splicing Languages with Two Restriction Enzymes. *ASM Science Journal*, 13, 1-7. https://doi.org/10.32802/asmscj.2020.sm26(5.5) **(Indexed by Scopus)**

8. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. (2021). Generalisations of Splicing Languages in DNA Splicing Systems Involving Two Palindromic Restriction Enzymes. *Malaysian Journal of Fundamental and Applied Sciences*, 17(2), 128-138. https://doi.org/10.11113/mjfas.v17n2.1370 **(Indexed by ISI WOS)**

Indexed Conference Proceedings

1. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. (2020). Molecular Aspects on Generalisations of Splicing Languages. In *Simposium Kebangsaan Sains Matematik (SKSM27)* (pp. 060008(1-6)). AIP Conference Proceedings. https://doi.org/10.1063/5.0018377 **(Indexed by Scopus)**

Non-Indexed Conference Proceedings

1. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. (2018). The Mathematical Modelling of DNA Splicing System with Palindromic and Non-Palindromic Restriction Enzymes. In *International Conference on Applied Analysis and Mathematical Modeling (ICAAMM 2018)* (pp. 127-138).

2. Fong, W. H., **Ismail, N. I.** & Sarmin, N. H. (2018). Generalisations of DNA Splicing Languages with Palindromic and Non-Palindromic Restriction Enzymes using Automata. In *International Conference on Applied Analysis and Mathematical Modeling (ICAAMM 2018)* (pp. 119-126).

Papers presented in Conferences

1. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. Generalisations of Splicing Languages in DNA Splicing Systems Involving Two Palindromic Restriction Enzymes. *UTM-Emerging Scientist Conference (UTM-ESCon) 2018*, Pulai Spring Resort, Johor Bahru, Malaysia, 2-3 May 2018.

2. Fong, W. H., **Ismail, N. I.** & Sarmin, N. H. The Modelling of DNA Splicing Systems with Two Non-Palindromic Restriction Enzymes. *2nd Asia International Multidisciplinary Conference (AIMC 2018)*, Universiti Teknologi Malaysia, Johor Bahru, Malaysia, 12-13 May 2018.

3. Sarmin, N. H, **Ismail, N. I.** & Fong, W. H. Automata for DNA Splicing Systems with Palindromic and Non-Palindromic Restriction Enzymes. *The 2018 International Conference on Information Technology, Engineering, Science, and its Applications (ITES 2018)*, Royal Ambarrukmo Hotel, Yogyakarta, Indonesia, 1-2 August 2018.

4. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. The Modelling of DNA Splicing Systems with One Non-Palindromic Restriction Enzyme, *7th International Graduate Conference on Engineering, Science and Humanities (IGCESH 2018)*, Universiti Teknologi Malaysia, Johor Bahru, Malaysia, 13-15 August 2018.

5. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. Automata for DNA Splicing Languages with Two Restriction Enzymes. *Simposium Kebangsaan Sains Matematik (SKSM26)*, Kota Kinabalu, Sabah, Malaysia, 28-29 November 2018.

6. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. Computation of Splicing Languages from DNA Splicing System based on Sequences of Restriction Enzymes. *8th International Conference on Computer Science & Computational Mathematics (ICCSCM 2019)*, Langkawi, Kedah, Malaysia, 4-5 July 2019.

7. Fong, W. H., **Ismail, N. I.** & Sarmin, N. H. Automata for DNA Splicing Languages with Palindromic and Non-Palindromic Restriction Enzymes using

Grammars. *7th International Conference and Workshop on Basic and Applied Sciences (ICOWOBAS 2019)*, KSL Resorts, Johor Bahru, Malaysia, 16-17 July 2019.

8. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. Molecular Aspects on Generalisations of Splicing Languages. *Simposium Kebangsaan Sains Matematik (SKSM27)*, Hotel Tenera, Bangi, Selangor, Malaysia, 26-27 November 2019.

Papers presented in Workshop

1. **Ismail, N. I.**, Fong, W. H., & Sarmin, N. H. Modelling of Splicing Systems in DNA Computing. *Joint Workshop for Global Engineers in Asia 2018 (JWGEA2018)*, Ritsumeikan University, Japan, 23 − 28 July 2018.

Copyrights

1. Graphical User Interface of DNA Splicing Systems with One Palindromic Restriction Enzyme. *Copyright Universiti Teknologi Malaysia*, LY2018001318. Fong, W. H., Sarmin, N. H. & **Ismail, N. I.**, 12 April 2018.

2. Splicing Languages Generators using Rule-Detection Algorithm in DNA Computing. *Copyright Universiti Teknologi Malaysia*, LY2019007573. Fong, W. H., Sarmin, N. H. & **Ismail, N. I.**, 5 December 2019.

Awards

1. Bronze medal in *19th Industrial Art and Technology Exhibition (INATEX 2017)*. Graphical User Interface of DNA Splicing Systems with Restriction Enzyme. Dewan Sultan Iskandar, Universiti Teknologi Malaysia, Johor, 21 – 23 November 2017.

2. Consolation prize in *Malaysian Mathematical Sciences Society (PERSAMA) Award 2020 under Invention Category*. DNA Splicing Language Generator (DNASpliceGen). Online platform via Microsoft Teams, 29 July 2021.